

```

22      .SBTTL  GETURM - GET UNIBUS RUN MASK
23      ;+
24      ;*** - GETURM - GET UNIBUS RUN MASK
25      ;
26      ; INPUTS:
27      ; R5 = ADDRESS OF CONTROLLER TEMPLATE
28      ;
29      ; OUTPUTS:
30      ; CNTBLK+6 = UNIBUS RUN MASK (MAYBE)
31      ; R0-R3 = DESTROYED
32      ;
33      ; -
34
35      GETURM:
36      CLR     CNTBLK+6      ; ASSUME NO URM
37      CLR     CNTBLK+10     ; ...
38      MOV     C.PRM+10(R5),R0 ; GET THE URM
39      BEQ     R1            ; IF EQ, NONE
40      CLR     R1            ; INITIALIZE COUNT (CLEAR C-BIT)
41      MOV     #URMVAL,R3    ; POINT TO OUTPUT BUFFER
42      ROR     R0            ; GET FIRST RUN MASK
43      BCS     30$           ; IF CS, UBA PRESENT
44      INC     R1            ; INCREMENT COUNT
45      ASR     R0            ; CHECK IF NEXT BUS RUN PRESENT
46      BCS     30$           ; IF CS, YES
47      RNE     10$           ; IF NE, MORE TO DO
48
49      10$:
50      MOV     #'J,-1(R3)    ; SET END OF RUN MASKS
51      SUB     #URM,R3       ; CALCULATE LENGTH OF STRING
52      MOV     R3,CNTBLK+6   ; STORE LENGTH OF STRING
53      MOV     #URM,CNTBLK+10 ; AND ADDRESS OF STRING
54
55      20$:
56      RETURN
57
58      ; FOUND RUN MASK
59
60      30$:
61      MOV     R1,R2         ; COPY CURRENT COUNT
62      ASL     R2            ; CONVERT TO A WORD INDEX
63      MOV     URMTBL(R2),R2 ; GET ADDRESS OF BUS NAME
64      MOV     (R2)+(R3)+    ; STORE BUS NAME
65      MOV     (R2)+(R3)+    ; ...
66      MOV     (R2)+(R3)+    ; ...
67      MOV     (R2)+(R3)+    ; ...
68      MOV     #'<,>,(R3)+  ; PLUS DELIMITER
69      BR      10$          ; AND GET NEXT

```

```

LOCAL DATA
97          .SBTTL  LOCAL DATA
98          ;****
99          ; LOCAL DATA
100         ;****
101         .ENABL  LC
102         EXERR = 2
103 000000    002    103 106 ERRSUP: .ASCIZ <EXERR>/CFE -- Event logging not supported/
104 000044    012    114 157 HEADR: .ASCIZ <12>/Logging characteristics as of %Y %3Z/
105 000112    012    113 156 HEADR1: .ASCIZ <12>/Known logging characteristics as of %Y %3Z/
106 000166    012    040 040 SNKMSG: .ASCIZ <12>/ Logging = %VA/
107 000207    012    011 116 NOINFO: .ASCIZ <12>/ No information/
108 000230    040    040 123 STAMSG: .ASCIZ / State = %VA/
109 000246    040    040 105 EVTMSG: .ASCIZ / Events =/<12>
110 000262    117    156 ON: .ASCII /On/
111 000002    117    146 146 OIF: .ASCII /Off/
112 000264    117    156 OFFL: .ASCII /Monitor/
113 000003    115    157 156 MONITR: .ASCII /File/
114 000267    115    151 154 FILE: .ASCII /File/
115 000007    106    156 FILEL: .-FILE
116 000276    103    156 CONSLE: .ASCII /Console/
117 000004    103    156 CONSLE: .-CONSLE
118 000302    103    156 CONSLE: .-CONSLE
119 000007    103    156 CONSLE: .-CONSLE
120         .EVEN
121 000312    000001 BITNUM: .BLKB 1 ; BIT NUMBER FOR EVENT MASKS
122 000313    000002 FLAGS: .BLKB 1 ; FLAGS BYTE FOR DISPLAYING EVENT MASKS
123 000314    000003 PREVNT: .BLKB 1 ; STORAGE FOR PREVIOUS EVENT TYPE
124 000315    000004 STATE: .BLKB 1 ; LOGGING STATE WORD
125 000316    000005 MSGBLK: .BLKW 2 ; MESSAGE ARGUMENT BLOCK
126         ;
127         ; SINK TYPE TABLE
128         ;
129 000322    000001 SNKTBL: .WORD FF.CON ; LOGGING CONSOLE
130 000324    000002 .WORD FF.FIL ; LOGGING FILE
131 000326    000004 .WORD FF.MON ; LOGGING MONITOR
132 000330    000000 .WORD 0 ; END OF TABLE

```

SHOLOG CREATED BY MACRO ON 29-JUN-85 AT 05:26 PAGE 3 B 3

MACRO CROSS REFERENCE CREF 04.00

MACRO NAME REFERENCES

EPRINT	#4-86						
ERRORS	#4-75	7-191					
FLTDF\$	#4-62	4-64					
PRINT	#4-70	7-206	7-246	8-282	9-309	10-329	10-376

SHOMOD - LIST MODULE CHARACTERI MACRO V05.03b Saturday 29-Jun-85 05:26 Page 10
 XPRCUG - SHOW MODULE X25-PROTOCOL GROUP

```

461          .SBTTL  XPRCUG - SHOW MODULE X25-PROTOCOL GROUP
462          ;+
463          *** XPRCUG - SHOW MODULE X25-PROTOCOL GROUP
464          ;
465          INPUT      $CUG = PARSED GROUP ID STRING
466                   CUG$DF TEMPLATE(S)
467          ;
468          OUTPUT     THE GROUP CHARACTERISTICS ARE DISPLAYED.
469          ;
470          ;-
471
472          002774          XPRCUG:
473          002774 012705 000000G      MOV    #STMLST,R5      ; POINT TO TEMPLATE LISTHEAD
474          003000 011505          10$:    MOV    (R5),R5      ; GET NEXT TEMPLATE
475          003002 001476          BEQ     40$                ; IF EQ, END OF LIST
476          003004 122765 000000G 000000G      CMPB   #CS.DTE,C.STS(R5) ; TEST TEMPLATE TYPE
477          003012 001372          BNE     10$                ; IF NE, NOT DTE TEMPLATE
478          003014 010504          MOV     R5,R4                ; SCAN CUGS FOR THIS DTE
479          003016 011404          20$:    MOV     (R4),R4      ; NEXT TEMPLATE
480          003020 001767          BEQ     10$                ; IF EQ, NO MORE CUGS, THIS DTE
481          003022 122764 000000G 000000G      CMPB   #CS.DTE,C.STS(R4) ; IS THIS THE NEXT DTE ?
482          003030 001763          BEQ     10$                ; IF EQ, YES
483          003032 122764 000000G 000000G      CMPB   #CS.CUG,C.STS(R4) ; IS IT A GROUP TEMPLATE?
484          003040 001366          BNE     20$                ; IF NE, KEEP LOOKING
485          003042 032767 000000G 000000G      BIT     #XP.KNG,$OPRND ; KNOWN GROUPS ?
486          003050 001010          BNE     30$                ; IF NE, YES
487          003052 026764 000000G 000000G      CMP     $CUG,$UNAM(R4) ; COMPARE NAMES
488          003060 001356          BNE     20$                ; IF NE, NEXT TEMPLATE
489          003062 026764 000002G 000002G      CMP     $CUG+2,$UN-M+2(R4) ; COMPARE NAMES
490          003070 001352          BNE     20$                ; IF NE, NEXT TEMPLATE
491          003072 012700 000000'          30$:    MOV     #CUGBK,R0      ; POINT TO ARGUMENT BLOCK
492          003076 016420 000000G          MOV     C$UNAM(R4),(R0)+ ; COPY GROUP NAME
493          003102 016420 000002G          MOV     C$UNAM+2(R4),(R0)+ ; ...
494          003106 016420 000000G          MOV     C$UCUG(R4),(R0)+ ; GROUP NUMBER
495          003112 010502          MOV     R5,R2                ; COPY TEMPLATE POINTER
496          003114 062702 000000G          ADD     #D$TADR,R2    ; POINT TO DTE NAME
497          000005          .REPT    5
498          .MOV     (R2)+,(R0)+ ; COPY RAD50 DTE ID STRING
499          .ENDR
500          003132 012720 000001          MOV     #BLANKL,(R0)+ ; ASSUME NOT TYPE BILATERAL
501          003136 012720 002022'          MOV     #BLANK,(R0)+ ; NEED BLANK FOR $EDMSG QUIRK
502          003142 132764 000000G 000000G      BITB   #CUB$IL,C$UFLG(R4) ; CHECK TYPE
503          003150 001406          BEQ     35$                ; IF EQ, PRINT
504          003152 012760 000013 177774          MOV     #TYPBL,-4(R0) ; ELSE TYPE = BILATERAL
505          003160 012760 002007' 177776          MOV     #TYPB,-2(R0) ;
506          003166 012703 001170'          35$:    MOV     #CUGFT,R3 ; POINT TO FORMATTING TABLE
507          003172 004767 000634          CALL    PRINT        ; FORMAT AND PRINT THE INFORMATION
508          003176 000707          BR       20$                ; CONTINUE
509          003200 000207          40$:    RETURN
510

```

SHOMOD - LIST MODULE CHARACTERI MACRO V05.03b Saturday 29-Jun-85 05:26 Page 11
 XPRCUG - SHOW MODULE X25-PROTOCOL GROUP


```

900                                     .SBTTL  MOVOPT  SET UP OPTIONAL PARAMETERS
901                                     ;+
902                                     ;** MOVOPT - SET UP OPTIONAL PARAMETERS
903                                     ;
904                                     ; INPUT      R5 = TEMPLATE ADDRESS
905                                     ;           R3 = ARGUMENT BLOCK POINTER
906                                     ;
907                                     ; OUTP'JT    THE PARAMETER BLOCK IS SET UP. IF THE OPTION
908                                     ;           IS PRESENT IN THE TEMPLATE IT WILL BE SET IN THE
909                                     ;           PRINT OPTIONS WORD.
910                                     ;
911                                     ; CALL      JSR      R4,MOVOP
912                                     ;           .WORD  OFFSET,WORDCOUNT,PRINT OPTION,TEMPLATE OPTION
913                                     ;
914                                     ; REGISTERS  R1 AND R2 ARE CORRUPTED.
915                                     ;
916                                     ;
917 004632 MOVOP:
918 004632 MOV      (R4)+,R2      ; GET OFFSET OF DATA FIELD
919 004634 ADD      R5,R2        ; POINT TO DATA
920 004636 MOV      (R4)+,R1      ; GET COUNT
921 004640 MOV      (R2)+,(R3)+   ; MOVE DATA
922 004642 DEC      R1           ; COUNT DOWN
923 004644 BGT      10$,        ; UNTIL DATA MOVED TO ARGBLK
924 004646 MOV      (R4)+,R1      ; GET PRINT OPTION
925 004650 BIT      (R4)+,C.STS(R5) ; WAS IT SELECTED IN TEMPLATE ?
926 004654 BEQ      20$,        ; IF EQ, NO
927 004656 BIS      R1,OPTION    ; ELSE SELECT FOR PRINTING
928 004662 RTS      R4
929
930 000001                                     .END
  
```

```

91          ; LOCAL DATA
92          ;****
93
94          .NLIST BEX
95          .ENABL LC
96 000000    012    120    162 HEADR: .ASCIZ <12>/Process characteristics as of %Y %3Z/
97 000046    012    113    156 HEADR1: .ASCIZ <12>/known processes characteristics as of %Y %3Z/
98 000124    012    120    162 PROMSG: .ASCIZ <12>/Process = %R/
99 000142    040    040    123 01MSG: .ASCIZ / State = %VA/
100 000160    040    040    115 02MSG: .ASCIZ / Maximum lines = %M/
101 000205    040    040    115 03MSG: .ASCIZ / Maximum lines = %M, Maximum controllers = %M/
102 000264    117    156    ON: .ASCII /On/
103          000002    ONL=-ON
104 000266    103    154    145 CLEAR: .ASCII /Cleared/
105          000007    CLEARL=-CLEAR
106          .EVEN
107
108          MSGBLK: .BLKW 2          ; MESSAGE ARGUMENT BLOCK
109 000302          .PSECT
  
```

```

284 000646 012702 000334'      25$:  MOV    #MS1BLK,R2      ; POINT TO MESSAGE BLOCK
285 000652 016501 000010      MOV    C.ADD(R5),R1      ; GET THE NODE ADDRESS
286 000656 004767 000334      CALL   PUTADR          ; USE PHASE 4 STYLE ADDRESS
287 000662 012701 000252'      MOV    #P4NOD,R1        ; POINT AT INPUT STRING TABLE
288 000666 016522 000704      MOV    C.NAM(R5),(R2)+      ; STORE THE REMOTE NAME
289 000672 016522 000006      MOV    C.NAM+2(R5),(R2)+      ;
290 000676 012700 000000G      30$:  MOV    #S$BUFF,R0      ; POINT AT OUTPUT BUFFER
291 000702 012702 000334'      MOV    #MS1BLK,R2      ; POINT AT ARGUMENT LIST
292 000706 004767 000000G      CALL   $EDMSG          ; FORMAT NODE ADDRESS
293 000712 112720 000040      35$:  MOV    #40,(R0)+      ; TAB OVER ...
294 000716 120027 000017G      CMPB   R0,#$BUFF+NAMCOL ; ...
295 000722 003773              BLE     35$
296 000724 012701 000261'      MOV    #NODNM,R1        ; POINT AT CONTROL STRING FOR NAME
297 000730 004767 000000G      CALL   $EDMSG          ; FORMAT NAME
298 000734              PRINT   #S$BUFF      ; PRINT THE MESSAGE
299 000744 004767 000276      CALL   GETDLL          ; PRINT DOWNLINE LOAD DATABASE
300 000750 000713              BR      10$          ; CONTINUE
301
302      ; END OF TEMPLATES FOUND
303
304 000752 005700              40$:  IST     R0          ; WERE ANY OBJECTS FOUND ?
305 000754 001004              BNE     50$          ; IF NE, YES .. OKAY
306 000756 032767 000000G 000000G  BIT     #WC.NOD,$WILD      ; WAS THIS A WILD CARD SPECIFICATION ?
307 000764 001401              BEQ     101$         ; IF EQ, NO .. REMOTE NOT FOUND
308 000766 000207              50$:  RETURN      ; RETURN TO CALLER
309
310      ; ERROR CONDITIONS
311
312 000770              101$:  ERRORS #ERRNOD      ; FAILED TO FIND REMOTE TEMPLATE
313 001000 000207              RETURN

```

SHOREM CREATED BY MACRO ON 29-JUN-85 AT 05:27 PAGE 3 B 8

SYMBOL CROSS REFERENCE CREF 04.00

SYMBOL	VALUE	REFERENCES
		#4-219 #4-219 4-219 #4-219 4-219 #4-220 #4-220 4-220 #4-220
		4-220 #4-222 #4-222 4-222 #4-222 4-222 #4-223 #4-223 4-223
		#4-224 4-224
.CONER	= ***** GX	8-481

TMPBUF - PARSE BUFFER DEFINITIO MACRO V05.03b Saturday 29-Jun-85 05:28 ^{B 9}
Table of contents

4-	56	MACRO DEFINITIONS
5-	92	LOCAL DATA
6-	135	TPARS STATE TABLES
7-	190	\$TMPBUF - BUILD BUFFER TEMPLATE
8-	234	SETCBN - SET TO CHANGE NUMBER
9-	265	SETCBS - SET CCB SIZE
10-	295	SETRBN - SET RDB NUMBER
11-	328	SETRBS - SET RDB SIZE
12-	359	SETSBN - SET SDB NUMBER
13-	388	SETSBS - SET SDB SIZE
14-	418	SETTHR - SET RDB THRESHOLD

```

449      .SBTTL  UPDALC  -- UPDATE ALLOCATION
450      :
451      :+
452      : ** UPDALC - UPDATE ALLOCATION
453      :
454      : INPUTS      RO,R1  VALUES TO BE MULTIPLIED
455      :
456      : OUTPUT      THE MULTIPLIED VALUE IS APPLIED TO
457      :              THE DOUBLE PRECISION BLOCK AREA MINIMUM
458      :
459      : UPDALC:
460      : CALL      $MUL      ; MULTIPLY EM OUT
461      : ADD      R1,$BLKMN  ; UPDATE LOW ORDER
462      : ADC      $BLKMN+2   ; AND HIGH ORDER
463      : ADD      RO,$LKMN+2 ; ...
464      :
465      : RETURN
466      :
467      .END
  
```

000001

TMPCN - PARSE OUTGOING CHANNEL MACRO V05.03b Saturday 29-Jun-85 05:28 Page 8
 SETLO - SET LOW END OF CHANNEL RANGE IN TEMPLATE

```

202 .SBTTL SETLO - SET LOW END OF CHANNEL RANGE IN TEMPLATE
203 .SBTTL SETHI - SET HIGH END OF CHANNEL RANGE IN TEMPLATE
204 ;+
205 ; ** SETLO - SET LOW END OF CHANNEL RANGE IN TEMPLATE
206 ; ** SETHI - SET HIGH END OF CHANNEL RANGE IN TEMPLATE
207 ;
208 ; INPUTS: .TPARS CONVENTIONS.
209 ;
210 ; OUTPUT: CHANNEL NUMBER SET IN TEMPLATE.
211 ;
212 ; -
213 ;
214 000144 SETLO: $GTNUM #C$HLO,#1,#4095.
215
216 000164 SETHI: $GTNUM #C$HHI,#1,#4095.,#CH$HI
217
218
219 000001 .END

```

TMPCN - PARSE OUTGOING CHANNEL MACRO V05.03b Saturday 29-Jun-85 05:28 Page 8-1
 Symbol table

```

242 .SBTTL SETPRI - SET THE PRIORITY
243
244 :+
245 :*** - SETPRI - SET THE CONTROLLER PRIORITY
246 :
247 :THIS ACTION ROUTINE CHECKS THE CONTROLLER PRIORITY TO MAKE SURE IT IS VALID
248 :AND STORES IT IN THE TEMPLATE.
249 :
250 :INPUT:
251 :.PNUMB = PRIORITY
252 :.PNUMH = PRIORITY (HIGH ORDER)
253 :TEMP = TEMPLATE ADDRESS
254 :
255 :OUTPUT:
256 :THE PRIORITY IS STORED IN C.PRI OF THE TEMPLATE.
257 :-
258
259 SETPRI:
260 000222      MOV     TEMP,RO      ; GET START OF TEMPLATE
261 000222 016700 177552      MOV     .PNUMB,C.PRI(RO) ; STORE THE PRIORITY IN THE TEMPLATE
262 000234 022767 000004 000012      CMP     #4,.PNUMB ; IS THE PRIORITY IN RANGE ?
263 000242 101004      BHI     10$ ; IF HJ, NO - REJECT
264 000244 022767 000006 000000G      CMP     #6,.PNUMB ; IS IT STILL IN RANGE ?
265 000252 103002      BHS     20$ ; IF HIS, YES - OKAY
266 000254 062716 000002      ADD     #2,(SP) ; ELSE, REJECT THE TRANSITION
267 000260 000207      10$:      RETURN
267 000260 000207      20$:
  
```


8 13
TMPCUG - PARSE CLOSED USER GROU MACRO V05.03b Saturday 29-Jun-85 05:29 Page 7-1
SETFLG - SET FLAGS BYTE

249 000001 .END

```

231 .SBTTL SETPRI - SET THE PRIORITY
232
233 ;*
234 ;*** - SETPRI - SET THE PROCESS PRIORITY
235 ; THIS ACTION ROUTINE CHECKS THE DDM PRIORITY TO MAKE SURE IT IS VALID
236 ; AND STORES IT IN THE TEMPLATE.
237
238 INPUT:
239 .PNUMB = PRIORITY
240 .PNUMH = PRIORITY (HIGH ORDER)
241
242 OUTPUT:
243 THE PRIORITY IS STORED IN C.PRI OF THE TEMPLATE.
244
245 ;
246
247 000140 SETPRI:
248 000140 016700 177636 MOV TEMP,RO ; GET START OF TEMPLATE
249 000144 016760 000000G 000010 MOV .PNUMB,C.PRI(RO) ; STORE THE PRIORITY IN THE TEMPLATE
250 000152 022767 000004 000000G CMP #4,.PNUMB ; IS THE PRIORITY IN RANGE ?
251 000160 101004 BHI 10$ ; IF HI, NO - REJECT
252 000162 022767 000006 000000G CMP #6,.PNUMB ; IS IT STILL IN RANGE ?
253 000170 103002 BHIS 20$ ; IF HIS, YES - OKAY
254 000172 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
255 000176 000207 20$: RETURN
  
```

```

169                                     .SBTTL  SETLKS - NUMBER OF LOGICAL LINKS PROCESS SHOULD SUPPORT
170
171                                     ;*
172                                     *** - SETLKS - SETUP NUMBER OF LOGICAL LINKS
173                                     THIS ACTION ROUTINE SETS UP THE NUMBER OF LOGICAL LINKS FOR THIS PROCESS.
174
175                                     INPUT:
176                                     .PNUMB = NUMBER OF LOGICAL LINKS
177                                     .PNUMH = (HIGH ORDER)
178
179                                     OUTPUT:
180                                     THE NUMBER OF LOGICAL LINKS ARE STORED IN C.LKS OF THE TEMPLATE
181
182                                     ;
183
184 000056                                SETLKS:
185 000056 016700 177716                MOV     TEMP,R0          ; GET START OF TEMPLATE
186 000062 016760 000000G 000010        MOV     .PNUMB,C.LKS(R0) ; STORE THE NUMBER OF LOGICAL LINKS
187 000070 005767 000000G                TST     .PNUMH          ; IS IT ONLY A WORD VALUE ?
188 000074 001003                        BNE     10$              ; IF NE, NO - REJECT
189 000076 105767 000001G                TSTB    .PNUMB+1        ; IS IT A BYTE VALUE ?
190 000102 001402                        BEQ     20$              ; IF EQ, YES - OKAY
191 000104 062716 000002                ADD     #2,(SP)          ; ELSE, REJECT TRANSITION
192 000110 000207                10$:
                                20$:    RETURN

```

TMPDEC CREATED BY MACRO ON 29-JUN-85 AT 05:31 PAGE 2 B 16
SYMBOL CROSS REFERENCE CREF 04.00

SYMBOL	VALUE	REFERENCES
		11-291 11-295 12-317 12-320 13-342 13-345 14-368 14-371 15-393
		15-396 16-418 16-422 17-443
.PNUMH	= ***** GX	7-137 8-215 9-241 10-267 11-293 12-318 13-343 14-369 15-394
.TPARS	= ***** GX	16-420 17-446
		6-163

```

566 .SBTTL GETSLT - GET THE LOADING INFORMATION CO-ROUTINE
567 *
568 *** - GETSLT - GET THE LOADING INFORMATION CO-ROUTINE
569
570 THIS CO-ROUTINE IS CALLED UNTIL A C-BIT IS SET. THIS CO-ROUTINE RETURNS
571 TRIBUTARY STATUS INFORMATION FOR EACH TRIBUTARY.
572
573 NOTE - THIS CO-ROUTINE USES R5 AS TEMPORARY STORAGE, THEREFORE THE CALLING
574 ROUTINE MUST NOT DESTROY R5.
575
576 INPUTS:
577 $DVNAM = LINE DESCRIPTOR BLOCK
578
579 OUTPUT:
580 C-BIT = SUCCESS/FAILURE
581 STABLK = THE LINE STATE ARGUMENT BLOCK
582 X2PBLX = X.25 LEVEL 2 PARAMETERS (IF LINE REQUIRES X.25 LEVEL 2)
583
584 -
585
586 GETSLT: MOV (SP),-(SP) ; MOVE RETURN ADDRESS UP ON STACK
587 003002 011646 MOV R5,2(SP) ; SAVE R5
588 003004 010566 000002 MOV CURSLT,R5 ; POINT AT CURRENT SLT
589 003010 016705 175062 CALL $FNLSLT ; FIND THE NEXT SLT TEMPLATE
590 003014 004767 000000G BCS 60$ ; IF CS, NOT FOUND
591 003020 103573
592
593 ; FOUND IT !!
594
595 10$: MOV R5,CURSLT ; SAVE CURRENT SLT ADDRESS
596 003026 005067 176430 CLR STABLK+4 ; ASSUME POINT TO POINT LINE
597 003032 005067 176430 CLR STABLK+10 ; ...
598 003036 005067 176426 CLR STABLK+12 ; ... MUST ZERO THE ADDRESS
599
600 ; IF LINE NEEDS LEVEL 2, GET PARAMETERS
601
602 2 003042 032765 000000G 000000G 15$: BIT #LF.X2P,$$FLG(R5) ; TEST FOR LEVEL 2
603 003050 001462 BEQ 30$ ; IF EQ, NOT X25 OR KMX
604 003052 012700 000000G MOV #TMLST,R0 ; POINT TO TEMPLATE LISTHEAD
605 003056 011000 20$: MOV (R0),R0 ; POINT TO NEXT TEMPLATE 'N LIST
606 003060 001446 BEQ 25$ ; IF EQ, NO LEVEL 2 TEMPLATE
607 003062 122760 000000G 000000G CMPB #CS.X2P,C.STS(R0) ; IS THIS A LEVEL 2 TEMPLATE
608 003070 001372 BNE 20$ ; IF NE, KEEP LOOKING
609 003072 026760 000000G 000000G CMP $DVNAM,X$2LIN(R0) ; IS THIS THE RIGHT LEVEL 2 TEMPLATE ?
610 003100 001366 BNE 20$ ; IF NE, NO
611 003102 026760 000002G 000002G CMP $DVNAM+2,X$2LIN+2(R0) ; DOES THE CONTROLLER MATCH ?
612 003110 001362 BNE 20$ ; IF NE, KEEP LOOKING
613 003112 026760 000004G 000004G CMP $DVNAM+4,X$2LIN+4(R0) ; DOES THE UNIT MATCH ?
614 003120 001366 BNE 20$ ; IF NE, KEEP LOOKING
615 003122 052767 000020 174750 31S #OP.X2P,DSPOPT ; SELECT OPTIONAL DISPLAY
616 003130 016067 000000G 176330 MOV X$2HBT(R0),X2PBLK ; GET HOLDBACK TIMER
617 003136 016567 000000G 176324 MOV $SLCTT(R5),X2PBLK+2 ; COUNTER TIMER
618 003144 011067 000000G 176326 MOV X$2BSZ(R0),X2PBL2 ; MAX DATA
619 003152 016067 000000G 176322 MOV X$2WSZ(R0),X2PBL2+2 ; MAX WINDOW
620 003160 016067 000000G 176316 MOV X$2RTC(R0),X2PBL3 ; RETRANSMIT COUNT
621 003166 016067 000000G 176312 MOV X$2RTT(R0),X2PBL3+2 ; RETRANSMIT TIMER
622 003174 000410 BR 30$

```

```

134          .SBTTL LOCAL SYMBOL DEFINITIONS
135          ****
136          : LOCAL SYMBOL DEFINITIONS
137          ****
138
139          000100      MSKLEN = 64.          ; NUMBER OF BITS IN EVENT MASKS
140
141          : FLAGS BIT DEFINITIONS
142
143          000001      F.STRT = 1          ; FORMATTING HAS BEEN STARTED
144          000002      F.PREV = 2          ; PREVIOUS BIT WAS SET
145          000004      F.NINF = 4          ; NO INFORMATION FOUND FOR SINK TYPE
146
147          : ASCII CHARACTER DEFINITIONS
148          :
149          000012      CR = 12
150          000011      TAB = 9
151          000055      DASH = '-'
152          000054      COMMA = ','
153          000056      PERIOD = '.'
154
155          : DEFINE EVENT TEMPLATE
156          :
157          000040      .ASECT
158          000000      .=0
159          000000      .BLKW 1          ; C.LNK
160          000002      .BLKW 1          ; C.STS
161          000004      .TMPLOC=.
162          000004      C.CLS: .BLKW 1          ; EVENT CLASS
163          000006      C.EVT: .BLKW 4          ; EVENT MASKS
164          000016      C.SNK: .BLKW 1          ; SINK TYPE
165          000020      C.LEN=.
166
167          : DEFINE LOGGING STATE TEMPLATE
168          :
169          000004      .=TMPLOC
170          000004      C.STA: .BLKB 1          ; LOGGING STATE
171          000032      C.LNT=.
172          .PSECT
173

```

```

SSSSSSSS HH HH 000000 MM MM 000000 DDDDDDDD
SSSSSSSS HH HH 000000 MM MM 000000 DDDDDDDD
SS HH HH 00 00 MMMM MMMM 00 00 DD DD
SS HH HH 00 00 MMMM MMMM 00 00 DD DD
SS HH HH 00 00 MM MM MM 00 00 DD DD
SS HH HH 00 00 MM MM MM 00 00 DD DD
SSSSSS HHHHHHHHHH 00 00 MM MM 00 00 DD DD
SSSSSS HHHHHHHHHH 00 00 MM MM 00 00 DD DD
SS HH HH 00 00 MM MM 00 00 DD DD
SS HH HH 00 00 MM MM 00 00 DD DD
SS HH HH 00 00 MM MM 00 00 DD DD
SS HH HH 00 00 MM MM 00 00 DD DD
SSSSSS HH HH 000000 MM MM 000000 DDDDDDDD
SSSSSS HH HH 000000 MM MM 000000 DDDDDDDD

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSSSS TT
LL SSSSSSSS TT

```

```

512 .SBTTL $SHXAC - SHOW MODULE X25-ACCESS
513 ;+
514 ; ** $SHXAC - SHOW MODULE X25-ACCESS
515 ;
516 ; INPUTS      $OPRND = TYPE OF REQUEST
517 ;
518 ; OUTPUT      X25-ACCESS PARAMETERS OUTPUT.
519 ; -
520
521 003202 $SHXAC::
522 003202 012705 003244' MOV #XACDTB,R5 ; POINT TO DISPATCH TABLE
523 003206 012704 000000G MOV #SHDRBK,R4 ; POINT TO ARGUMENT BLOCK
524 003212 012724 000012 MOV #XACL,(R4)+ ; MODULE IS X25-SERVER
525 003216 012724 001674' MOV #XAC,(R4)+ ;
526 003222 016703 000000G MOV $OPRND,R3 ; GET REQUEST TYPE
527 003226 042703 000000C BIC #^C<XA.DST!XA.KND>,R3 ; CLEAR DONT CARE BITS
528 003232 012767 177777 174552 MOV #-1,OPTIONS ; WANT ALL OUTPUT
529 003240 CALLR DISPAT ; DISPATCH TO ROUTINE
530 ;
531 ; DISPATCH TABLE
532 ;
533 ;
534 003244 003634' 000000 000017 XACDTB: .NLIST BEX
535 003254 003636' 000000G 000017 .WORD XACCHA, 0, CHARL, CHAR ; CHARACTERISTICS
536 003264 003636' 000000G 000022 .WORD SHOXAD, XA.DST, CHARL, CHAR ; DESTINATION
537 003274 000000 .WORD SHOXAD, XA.KND, KDSTL, KDST ; KNOWN DESTINATIONS
538 .WORD 0
539 .LIST BEX

```



```

BLANK 002022R      DSTB7 000034R      002 FL.MJX= ***** GX      P$SNAM= ***** GX      X$3MW = ***** GX
BLANKL= 000001R    DSTF1 001532R      FL.TRB= ***** GX      P$SPOR= ***** GX      X$3MXP= ***** GX
CHAR 001720R      DSTM1 001266R      FMTLN2 004126R      P$SDNAM= ***** GX      X$3RM = ***** GX
CHARL = 000017R    DSTM10 001505R      FMTTRN1 000255R      SCRBLK 000014R      X$3RT = ***** GX
CHNBK 001101R      DSTM2 001312R      FMTTRN1 000250R      SHODST 004252R      X$3SM = ***** GX
CHNBND= 000130R    DSTM3 001350R      FMTSRN 000265R      SHOXAD 003636R      X$3ST = ***** GX
CHNBUF 000034R      DSTM4 001367R      FMTSR1 000275R      SPCDST 002024R      X$3CT = ***** GX
CHNHDR 001064R      DSTM5 001405R      FMT2R5 000300R      TYPB 002007R      X$3MXC= ***** GX
CH$H1 = ***** GX  DSTM6 001433R      HURMSG 000213R      TYPBL = 000013      X29 001706R
CS.CHN= ***** GX  DSTM7 001462R      HDKPR1 000000R      XAC 001674R      X29CHA 003566R
CS.CUG= ***** GX  DS.CHI= 000100      KCUG 001773R      XACCHA 003634R      X29DTB 003534R
CS.DST= ***** GX  DS.CLD= 000040      KCUGL = 000014      XACDTB 003244R      X29L = 000012
CS.DTE= ***** GX  DS.CUG= 000020      KDST 001737R      XACL = 000012      X9.DST= ***** GX
CS.LLC= ***** GX  DS.MSK= 000200      KDSTL = 000022      XADBK 000000R      X9.KND= ***** GX
CS.MSK= ***** GX  DS.NUM= 000010      KDTE 001761R      XADB1 000000R      $BUFF = ***** GX
CS.NUM= ***** GX  DS.OPT= 000770      KDTEL = 000012      XADFT 001640R      $CBTA = ***** GX
CS.RDT= ***** GX  DS.VAL= 000400      KNOWN 000006R      XADM1 001574R      $CFERR= ***** GX
CS.SHI= ***** GX  DTEBK 000000R      L$LCNT= ***** GX      XA.DST= ***** GX      $CUG = ***** GX
CS.SLO= ***** GX  DTEB1 000000R      L$LCM= ***** GX      XA.KND= ***** GX      $C5TA = ***** GX
CS.UNI= ***** GX  DTEB2 000012R      L$LNAM= ***** GX      XPR 001646R      $DST = ***** GX
CS.VAL= ***** GX  DTEB3 000020R      L$NAM = ***** GX      XPRCHA 002156R      $DTE = ***** GX
CS.X29= ***** GX  DTEFT 001046R      L$STR1 = ***** GX      XPRCUG 002774R      $EDMSG= ***** GX
CS.X3P= ***** GX  DTEM1 000724R      L$UNT = ***** GX      XPRDTB 002104R      $FIAGS= ***** GX
CUGBK 000000R      002 DTEM2 000740R      MDVDP1 004632R      XPRDTE 002326R      $HDBRK= ***** GX
CLGB1 000000R      002 DTEM3 001001R      NOINF 000171R      XPRL = 000014      $OPRND= ***** GX
CUGB2 000004R      002 DTYPE 000010R      OPTIDN 000012R      XP.CUG= ***** GX      $PRINT= ***** GX
CUGFT 001170R      D$ACUG= ***** GX      PRCHBK 000000R      XP.DTE= ***** GX      $PSIPT= ***** GX
CUGM1 001116R      D$ANUM= ***** GX      PRCHB1 000000R      XP.KND= ***** GX      $SHXAC 003202RG
CUGM2 001134R      D$ASHI= ***** GX      PRCHB2 000004R      XP.KNG= ***** GX      $SHXPR 002042RG
CUG$BIL= ***** GX  D$ASLD= ***** GX      PRCHB3 000010R      XSV 001662R      002 $SHXSV 003276RG
C$HLO = ***** GX  D$CMSK= ***** GX      PRCHB4 000014R      002 $SVCBK 000000R      002 $SHX29 003464RG
C$CUG= ***** GX  D$CVAL= ***** GX      PRCHB5 000020R      002 $SVCF1 001202R      002 $TMLST= ***** GX
C$UFLG= ***** GX  D$SNAM= ***** GX      PRCHB6 000024R      002 $SVCF2 001260R      $SBASE= 000000R
C$UNAM= ***** GX  D$SOBJ= ***** GX      PRCHT 000672R      002 $SVCM1 001202R      002 $DBIT= 000004
C.STS = ***** GX  D$STSK= ***** GX      PRCHM1 000304R      002 $SVDB1 003346R      $STTX0= 000001
D$ACUG= ***** GX  D$STYP= ***** GX      PRCHM2 000355R      002 $SVL = 000012      $STTX1= 000002
DISPAT 003750R      D$TADR= ***** GX      PRCHM3 000430R      XSVL = 000012      $STT1 = 000002
DSTBK 000000R      002 D$TCT = ***** GX      PRCHM4 000500R      X$.DST= ***** GX      $STT10= 000013
DSTB1 000000R      002 D$THSH= ***** GX      PRCHM5 000554R      X$.KND= ***** GX      $STT2 = 000003
DSTB10 000062R     002 D$TLIN= ***** GX      PRCHM6 000624R      X$3CT = ***** GX      $STT3 = 000005
DSTB2 000004R      002 D$TSTA= ***** GX      PRINT 004032R      X$3DFP= ***** GX      $STT4 = 000002
DSTB3 000012R      002 ERRREG 000130R      PRICHN 002560R      X$3DW = ***** GX      $STT5 = 000001
DSTB4 000024R      002 EX$ERR= 000002      X$3KM = ***** GX      $STT6 = 000001
DSTB5 000030R      002 FL.MTP= ***** GX      X$3KT = ***** GX      $STT7 = 000013
DSTB6 000032R
. ABS. 000000 000 (RW,I,GBL,ABS,DVR)
004664 001 (RW,I,LCL,REL,CCN)
ARGBLK 000110 002 (RW,D,LCL,REL,DVR)
Errors detected: 0

```

*** Assembler statistics

Work file reads: 0
Work file writes: 0
Size of work file: 9311 words (37 Pages)

```

111 .SBTTL $SHOPRO - SHOW PROCESS CHARACTERISTICS
112
113 *** - $SHOPRO - SHOW PROCESS CHARACTERISTICS
114
115 INPUT:
116 $PRNAM = THE PROCESS NAME ( ZERO FOR 'KNOWN PROCESSES' )
117
118 OUTPUT:
119 THE CHARACTERISTICS FOR THE PROCESS IS DISPLAYED.
120
121 -
122
123 $SHOPRO::
124 000302 012701 000000' MOV #HEADR,R1 ; ASSUME ONLY 1 PROCESS
125 000306 032767 000000G 000000G BIT #WC.NAM,$WILD ; FOR ALL PROCESSES ?
126 000314 001402 5$ BEQ 5$ ; IF EQ, NO .. OKAY
127 000316 012701 000046' MOV #HEADR1,R1 ; ELSE, DIFFERENT HEADER
128 000322 012700 000000G 5$ MOV #S$BUFF,R0 ; POINT AT OUTPUT BUFFER
129 000326 012702 000000G MOV #S$GTBUF,R2 ; POINT AT ARGUMENT BLOCK
130 000332 004767 000000G CALL $EDMSG ; FORMAT THE HEADER
131 000336 PRINT #S$BUFF ; PRINT HEADER
132 000346 012705 000000G MOV #S$TMLST,R5 ; GET START OF TEMPLATE LIST
133
134 SEARCH FOR THE PROCESS NAME
135
136 000352 010500 10$ MUV R5,R0 ; COPY CURRENT TEMPLATE ADDRESS
137 000354 004767 000000G CALL $FNPRO ; FIND THE NEXT PROCESS
138 000360 103517 BCS 60$ ; IF CS, NO MORE
139 000362 010005 MOV R0,R5 ; SAVE NEW TEMPLATE
140
141 DISPLAY THE CHARACTERISTICS
142
143 000364 012702 003276' MOV #MSGBLK,R2 ; POINT R2 AT ARGUMENT BLOCK
144 000370 016512 000004' MOV C.NAM(R5),(R2) ; STORE THE PROCESS NAME
145 000374 012700 000000G MOV #S$BUFF,R0 ; POINT R0 AT OUTPUT BUFFER
146 000400 012701 000124' MOV #PROMSG,R1 ; POINT R1 AT INPUT STRING
147 000404 004767 000000G CALL $EDMSG ; EDIT THE STRING
148 000410 PRINT #S$BUFF ; PRINT THE LINE
149 000420 012702 000276' MOV #MSGBLK,R2 ; POINT R2 AT ARGUMENT BLOCK
150 000424 012712 000002' MOV #ONL,(R2) ; ASSUME PROCESS IS ON
151 000430 012762 000264' 000002' MOV #ONL,2(R2)
152 000436 032765 000000G 000006' BIT #ZF.MFL,C.FLG(R5) ; IS PROCESS ON ?
153 000444 001005 30$ BNE 30$ ; IF NE, YES .. OKAY
154 000446 012712 000007' MOV #CLEARL,(R2) ; ELSE, MUST BE CLEARED
155 000452 012762 000265' 000002' MOV #CLEAR,2(R2)
156 000460 012700 000000G 30$ MOV #S$BUFF,R0 ; GET THE INPUT STRING
157 000464 012701 000142' MOV #O1MSG,R1 ; POINT R1 AT THE FORMAT STRING
158 000470 004767 000000G CALL $EDMSG ; EDIT THE STRING
159 000474 PRINT #S$BUFF ; PRINT THE STRING
160 000504 012702 000276' MOV #MSGBLK,R2 ; POINT AT ARGUMENT BLOCK
161 000510 005012 L R (R2) ; GET LINE TABLE COUNT WITHOUT SIGN EXTEND
162 000512 122765 000000G 000000G CMB #CS.LLC,C.STS(R5) ; IS THIS AN LLC PROCESS ?
163 000520 001015 50$ BNE 50$ ; IF NE, NO .. LDM/DLC
164
165 LLC LINE TABLE COUNT
166
167 000522 156512 000014 BLSB C.NLT(R5),(R2) ; GET LINE TABLE COUNT

```

315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342

001002
012702 000334'
016701 000000G
007767 000200
012701 000252'
016722 000000G
016722 000000G
012700 000000G
012702 000334'
004767 000000G
112720 000040
120027 000017G
003773
012701 000265'
004767 000000G
001070
001100 000207

*** - EXENOD - PRINT EXECUTOR NODE INFORMATION

INPUTS:
NONE.

OUTPUTS:
INFORMATION IS PRINTED.

EXENOD:

MOV #MS1BLK,R2 ; AND AT ARGUMENT BLOCK
 MOV \$NTADD,R1 ; GET EXEC ADDRESS
 CALL PUTADR ; PUT PHASE 4 ADDRESS
 MOV #P4NOD,R1 ; USE AREA FORMAT NODE ADDRESS
 MOV \$NTNAM,(R2)+ ; AND THE NODE NAME
 MOV \$NTNAM+2,(R2)+ ; ...
 MOV # \$BUFF,R0 ; POINT AT OUTPUT BUFFER
 MOV #MS1BLK,R2 ; AND AT ARGUMENT BLOCK
 CALL \$EDMSG ; FORMAT THE STRING
 35\$: MOVB #40,(R0)+ ; TAB OVER ...
 CMPB R0,# \$BUFF+NAMCOL ; ...
 BLE 35\$; ...
 MOV #EXENM,R1 ; EXECUTOR NODE FORMATTING STRIN
 CALL \$EDMSG ; FORMAT IT
 PRINT # \$BUFF ; PRINT THE MESSAGE
 RETURN

SHOREM CREATED BY MACRO ON 29-JUN-85 AT 05:27 PAGE 4 C 8
MACRO CROSS REFERENCE CREF 04.00

MACRO NAME REFERENCES

ASL\$	#4-104	10-611	10-619						
ASR\$	#4-104								
DLLDAT	#4-98	4-208	4-209	4-211	4-212	4-213	4-215	4-217	4-218
	4-220	4-222	4-223	4-224					4-219
EPRINT	#4-78								
ERRR\$	#4-67	4-312							
PRINT	#4-60	4-298	5-341	6-372	6-375	8-454	8-478		
RESRG	#4-104	9-497	10-638						
SAVRG	#4-104	9-494	10-634	10-636					

.TITLE TMPBUF - PARSE BUFFER DEFINITION IN CETAB
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE BUFFER DEFINITION

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-DEC-79
DECNET-11M/S V3.0
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82
DECNET-11M V3.1
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83
DECNET-11M V4.0
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85
DECnet-11M/S V4.2
DECnet-11M-Plus V3.0
DECnet-Micro/RSX V1.0

TMPBUF - PARSE BUFFER DEFINITIO MACRO V05.03b Saturday 29-Jun-85 05:28 Page 15-1
Symbol table

BUFKTB 000000RG 003 HICBSZ 000010R
BUFSB 000000RG 002 HIRBNM 000014R
CS.BUF= ***** GX
C.CBNM 000004 HIRBSZ 000020R
C.CBSZ 000006 HIRBSZ 000026R
C.LEN= 000022 HIRBSZ 000032R
C.RBNM 000010 LOCBNM 000002R
C.RBSZ 000012 LOCBNM 000006R
C.SBNM 000014 LOCBNM 000012R
C.SBSZ 000016 LOCBNM 000016R
C.SIS= ***** GX
C.THSZ 000020 LOCBNM 000022R
HICBNM 000004R LOCBNM 000024R
LOCBNM 000030R

SETCBN 000134R
SETCBS 000176R
SETRBN 000234R
SETRBS 000306R
SETSBN 000350R
SETSBS 000404R
SETTHR 000442R
TEMP 000000R
\$ALPHA= 000022
\$ANY = 000020
\$BLANK= 000006
\$CCBNM= ***** GX

\$DIGIT= 000024
\$DNUMB= 000014
\$EOS = 000012
\$ERROR= ***** GX
\$EXIT = 000000
\$FAIL = 177777
\$LAMBDA= 000000
\$LIMITB 000002RG
\$MINSB= ***** GX
\$NUMB= 000002
\$RAD50= 000016
\$RDBNM= ***** GX

\$RDBSZ= ***** GX
\$SDBNM= ***** GX
\$STRNG= 000004
\$SURXP= 000010
\$TALOC= ***** GX
\$THRSZ= ***** GX
\$TMBUF 000034RG
\$\$\$FLG= 177777
\$\$\$KEY= 177777
\$\$\$STA= 000000
\$PNUMB= ***** GX
\$TPARS= ***** GX

. ABS. 000022 000 (RW,I,GBL,ABS,OVR)
000506 001 (RW,I,LCL,REL,CON)
\$STATE 000060 002 (RW,D,LCL,REL,CON)
\$KTAB 000000 003 (RW,D,LCL,REL,CON)
\$KSTR 000000 004 (RW,D,LCL,REL,CON)
Errors detected: 0

*** Assembler statistics

Work file reads: 0
Work file writes: 0
Size of work file: 10633 Words (42 Pages)
Size of core pool: 14440 Words (55 Pages)
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:26.15
SY:TMPBUF.V2,[132,134]TMPBUF/CR/-SP=SY:[1,1]RSXMC.M.SML/ML,[130,110]NETLIB/ML,[132,10]TMPBUF

IMPCHN - PARSE OUTGOING CHANNEL MACRO V05.03b Saturday 29-Jun-85 05:28 ^{C 11} Page 8-1
 Symbol table

CHNKTB 000000RG	003	EX\$SEV= 000004	\$DIGIT= 000024	\$LAMDA= 000000	\$\$\$FLG= 177777
CHNSTB 000000RG	002	SETHI 000164R	\$DNUMB= 000014	\$NUMBR= 000002	\$\$\$KEY= 177777
CH\$HI = ***** GX		SETLO 000144R	\$EOS = 000012	\$RAD50= 000016	\$\$\$STA= 000000
CS.CHN= ***** GX		\$ALPHA= 000022	\$ERROR= ***** GX	\$STRNG= 000004	.ERROR= ***** GX
C\$HHI = ***** GX		\$ANY = 000020	\$EXIT = 000000	\$SUBXP= 000010	.FLAGS= ***** GX
C\$HLEN= ***** GX		\$BLANK= 000006	\$FAIL = 177777	\$TALOC= ***** GX	.TEMP = ***** GX
C\$HLO = ***** GX		\$CFERR= ***** GX	\$GTNUM= ***** GX	\$TMCHN 000040RG	.TPARS= ***** GX
ERRCHN 000000R		\$CLERR= ***** GX			

. ABS. 000000 000 (RW,I,GBL,ABS,OVR)
 000212 001 (RW,I,LCL,REL,CON)
 \$STATE 000032 002 (RW,D,LCL,REL,CON)
 \$KTAB 000000 003 (RW,D,LCL,REL,CON)
 \$KSTR 000000 004 (RW,D,LCL,REL,CON)
 Errors detected: 0

*** Assembler statistics

Work file reads: 0
 Work file writes: 0
 Size of work file: 10992 Words (43 Pages)
 Size of core pool: 14440 Words (55 Pages)
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:14.10
 SY: TMPCHN.V2,[132,134]IMPCHN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]IMPCHN

```

269                                     .SBTTL  SETURM - SET THE UNIBUS RUN MASK
270                                     ;+
271                                     *** - SETURM - SET UNIBUS RUN MASK
272                                     ;
273                                     INPUTS:
274                                     .PNUMB = THE UNIBUS RUN MASK
275                                     .PNUMH = THE UNIBUS RUN MASK (HIGH ORDER BITS)
276                                     TEMP = TEMPLATE ADDRESS
277                                     ;
278                                     OUTPUTS:
279                                     THE UNIBUS RUN MASK IS STORED IN THE TEMPLATE
280                                     THE TRANSITION IS ACCEPTED/REJECTED
281                                     ;
282                                     -
283
284 000262                                     SETURM:
285 000262 016700 177512                     MOV     TEMP,RO          ; GET THE TEMPLATE ADDRESS
286 000266 016760 000000G 000014             MOV     .PNUMB,C.URM(RO) ; STORE THE URM IN THE TEMPLATE
287 000274 005767 000000G                     TST     .PNUMH          ; IS THE URM A WORD VALUE ?
288 000300 001402                               BEQ     10$           ; IF EQ, YES - OKAY
289 000302 062716 000002                     ADD     #2,(SP)         ; ELSE REJECT THE TRANSITION
290 000306 000207 10$: RETURN
291
292
293 000001                                     .END

```


C 13

TMPCUG - PARSE CLOSED USER GROU MACRO V05.03b Saturday 29-Jun-85 05:29 Page 7-2

Symbol table

CS.CUG= ***** GX	ERRFLG 000066R	\$BLANK= 000006	\$FAIL = 177777	\$TALOC= ***** GX
CS.FLG= ***** GX	ERRNAM 000000R	\$BYTMN= ***** GX	\$GTNMB= ***** GX	\$TMCUG 000116RG
CUGKTB 000000RG	003 EX\$SEV= 000004	\$CFERR= ***** GX	\$GTNUM= ***** GX	\$\$\$FLG= 177777
CUGSTB 000000RG	002 G\$LENX= ***** GX	\$CLERR= ***** GX	\$GTR50= ***** GX	\$\$\$KEY= 177777
CUG2 000040R	002 SETCUG 000244R	\$DIGIT= 000024	\$LAMBDA= 000000	\$\$\$STA= 000000
C.CUG 000010	SETFLG 000264R	\$DNUMB= 000014	\$NUMBER= 000002	.ERROR= ***** GX
C.FLG 000012	SETNAM 000230R	\$EOS = 000012	\$RAD50= 000016	.FLAGS= ***** GX
C.LEN = 000013	\$ALPHA= 000022	\$ERROR= ***** GX	\$STRNG= 000004	.TEMP = ***** GX
C.NAM 000004	\$ANY = 000020	\$EXIT = 000000	\$SUBXP= 000010	.TPARS= ***** GX
ERRCUG 000031R				

. ABS. 000013	000	(RW,I,GBL,ABS,OVR)
000312	001	(RW,I,LCL,REL,CON)
\$STATE 000066	002	(RW,D,LCL,REL,CON)
\$KTAB 000000	003	(RW,D,LCL,REL,CON)
\$KSTR 000000	004	(RW,D,LCL,REL,CON)

Errors detected: 0

*** Assembler statistics

Work file reads: 0
 Work file writes: 0
 Size of work file: 10984 Words (43 Pages)
 Size of core pool: 14440 Words (55 Pages)
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:20.34
 SY: TMPCUG.V2,C132,134JTMPCUG/CR/-SP=SY:C1,1JRSXMCML.SML/ML,C130,110JNETLIB/ML,C132,10JTMPCUG

```

257      .SBTTL SETNLT - STORE NUMBER OF LINE TABLES
258      :+
259      :*** - SETNLT - STORE NUMBER OF LINE TABLES
260      :
261      :INPUT:
262      :.PNUMB = NUMBER OF LINE TABLE TO BE SET IN PROCESS SPACE
263      :
264      :OUTPUT:
265      :NUMBER IS VALIDATED AND THE TRANSITION IS ACCEPTED/REJECTED
266      :
267      :-
268
269 000200      SETNLT:
270 000200      MOV     TEMP,R0          ; GET THE TEMPLATE ADDRESS
271 000204      MOVB    .PNUMB,C.NLT(R0) ; STORE THE LINE COUNT
272 000212      CMP     .PNUMB,#$MXNLT  ; VALUE IN RANGE ?
273 000220      BLOS    10$             ; IF LOS, YES
274 000222      ADD     #2,(SP)         ; ELSE, REJECT
275 000226      10$:  RETURN
  
```

194
 195
 196
 197
 198
 199
 200
 201
 202
 203
 204
 205
 206
 207
 208
 209

```
.SBTTL SETNDC - NUMBER OF NODE COUNTERS
*** - SETNDC - SETUP NUMBER OF NODE COUNTERS
THIS ACTION ROUTINE SETS UP THE NUMBER OF NODE COUNTERS FOR THIS PROCESS.

INPUT:
.PNUMB = NUMBER OF NODE COUNTERS
.PNUMH = (HIGH ORDER)

OUTPUT:
THE NUMBER OF NODE COUNTERS ARE STORED IN C.DNC OF THE TEMPLATE
THE SIZE OF THE NODE COUNTERS IS INCLUDED IN THE BLOCK AREA ALLOCATION.
```

210 000112
 211 000112 016700 177662
 212 000116 026727 000000G 000252
 213 000124 101011
 214 000126 016760 000000G 000006
 215 000134 005767 000000G
 216 000140 001003
 217 000142 105767 000001G
 218 000146 001402
 219 000150 062716 000002
 220 000154 000207

```
SETNDC:
MOV TEMP,PO ; GET START OF TEMPLATE
CMP .PNUMB,#MXNDC ; CHECK AGAINST MAXIMUM NODE COUNTERS
10$ BHI 10$ ; IF HI, TOO MANY
MOV .PNUMB,C.NDC(RO) ; STORE THE NUMBER OF NODE COUNTERS
TST .PNUMH ; IS IT ONLY A WORD VALUE ?
BNE 10$ ; IF NE, NO - REJECT
TSTB .PNUMB+1 ; IS IT A BYTE VALUE ?
BEQ 20$ ; IF EQ, YES - OKAY
10$ ADD #2,(SP) ; ELSE, REJECT TRANSITION
20$ RETURN
```

TMPDEC CREATED BY MACRO ON 29-JUN-85 AT 05:31 PAGE 3 C 16
 MACRO CROSS REFERENCE CREF 04.00

MACRO NAME REFERENCES

DBGTP\$	#5-69									
DECTP\$	#4-50	4-52								
ISTAT\$	#4-50	5-69								
MSGDF\$	#4-50	4-53								
MTRAN\$	#5-69									
STAT\$	#4-50	5-71	#5-74	#5-77	#5-80	#5-83	#5-86	#5-89	#5-92	#5-95
	#5-98	#5-101	#5-104	#5-107	#5-110	#5-113	#5-116	#5-119	#5-122	#5-125
	#5-128	#5-131	#5-136	#5-139						
RAN\$	#4-50	#5-72	#5-75	#5-78	#5-81	#5-84	#5-87	#5-90	#5-93	#5-96
	#5-99	#5-102	#5-105	#5-108	#5-111	#5-114	#5-117	#5-120	#5-123	#5-126
	#5-129	#5-132	#5-137							

```

623 003176 012700 000710' 25$: MOV #X2PLIN,R0 ; POINT TO LINE BUFFER
624 003202 004767 000224 CALL FMTLN2 ; FORMAT LINE ID
625 003206 ERROR$ ERRRX2P ; NO X2P$DF FOR LINE
626 003216 012767 000007 176232 30$: MOV #CLEARL,STABLK+0 ; ASSUME THIS DEVICE IS CLEARED
627 003224 012767 000745' 176226 MOV #CLEAR,STABLK+2 ; ...
628 003232 032765 000000G 000000G BIT #LF.MFL,$$LFLG(R5) ; IS IT CLEARED ?
629 003240 001420 BEQ 35$ ; IF EQ, YES .. OKAY
630 003242 012767 000003 176206 MOV #OFFL,STABLK+0 ; ELSE, ASSUME IT'S OFF
631 003250 012767 000742' 176202 MOV #OFF,STABLK+2 ; ...
632 003256 032765 000000G 000000G BIT #LF.ENA,$$LFLG(R5) ; IS IT ENABLED ?
633 003264 001406 BEQ 35$ ; BR IF NOT
634 003266 012767 000002 176162 MOV #ONL,STABLK+0 ; ELSE STATE = ON
635 003274 012767 000740' 176156 MOV #ON,STABLK+2 ; ...
636 003302 032767 000000G 000000G 35$: BIT #FL.MTP,$FLAGS ; IS THE LINE MULTI-POINT ?
637 003310 001413 BEQ 45$ ; IF EQ, NO
638 003312 032767 000000G 000000G BIT #FL.TRI,$FLAGS ; DO WE NEED TRIBUTARY INFO ?
639 003320 001421 BEQ 55$ ; IF EQ, NO
640 003322 004767 000000G CALL $FNSTA ; FIND THE NEXT STATION TEMPLATE
641 003326 103430 BCS 60$ ; IF CS, NO MORE STATIONS
642
643 003330 132765 000000G 000002G 4$: BITB #SF.ENA,C.PRM+2(R5) ; IS THE STATION OFF ?
644 003336 000403 BR 50$ ; CONTINUE
645 003340 032765 000000G 000002G 45$: BIT #LF.ENA,C.PRM+2(R5) ; IS THE LINE OFF ?
646 003346 001406 BEQ 50$ ; IF EQ, YES .. OKAY
647 003350 012767 000002 176100 MOV #ONL,STABLK+0 ; ELSE, IT'S ON
648 003356 012767 000740' 176074 MOV #ON,STABLK+2 ; ...
649 003364 000241 55$: CLC ; SUCCESS !!
650 003366 004736 CALL a(SP)+ ; CALL THE CALLER BACK
651
652 003370 032767 000000G 000000G BIT #FL.MTP,$FLAGS ; IS THE LINE MULTI-POINT ?
653 003376 001410 BEQ 65$ ; IF EQ, NO LEAVE
654 003400 032767 000000G 000000G BIT #FL.TRI,$FLAGS ; WANT ANY MORE TRIBUTARY INFO ?
655 003406 001205 BNE 10$ ; IF NE, YES
656 003410 016605 000002 60$: MOV 2(SP),R5 ; ELSE, RESTORE R5
657 003414 012616 MOV (SP)+,(SP) ; MOVE RETURN ADDRESS DOWN ON STACK
658 003416 000401 BR 70$ ; AND RETURN
659 003420 012605 65$: MOV (SP)+,R5 ; ELSE, RESTORE R5
660 003422 000261 70$: SEC ; ALL DONE
661 003424 000207 RETURN

```

```

175 .SBTTL SSHLOG - SHOW EVENT LOGGING CHARACTERISTICS
176
177 *** - SSHLOG - SHOW EVENT LOGGING CHARACTERISTICS
178
179 INPUT:
180 NONE.
181
182 OUTPUT:
183 THE EVENT LOGGING CHARACTERISTICS ARE DISPLAYED.
184
185
186
187 SSHLOG::
188 000332 012705 000000G MOV #STMLST,R5 ; GET START OF TEMPLATE LIST
189 000336 011505 10$: MOV (R5),R5 ; GET NEXT ONE IN LIST
190 000340 0C1005 BNE 20$ ; BR IF FOUND ONE
191 000342 ERROR$ #ERRSUP ; INDICATE EVENT LOGGING NOT SUPPORTED
192 000352 000521 BR 90$ ; AND EXIT
193 000354 122765 000000G 000000G 20$: CMPB #CS.LOG,C.STS(R5) ; IS THIS A LOGGING STATE TEMPLATE?
194 000362 001365 BNE 10$ ; BR IF NO
195 000364 116567 000004 177723 MOVB C.STA(R5),STATE ; SAVE LOGGING STATE WORD
196
197 ; PRINT HEADER MESSAGE
198
199 000372 012701 000044' MOV #HEADR,R1 ; ASSUME LOGGING SINK TYPE
200 000376 032767 000000G 000000G BIT #WC.LOG,$WILD ; ALL MONITORS ?
201 000404 001402 BEQ 30$ ; IF EQ, NO ... OKAY
202 000406 012701 000112' MOV #HEADR1,R1 ; ELSE, DIFFERENT HEADER
203 000412 012700 000000G 30$: MOV #B$UFF,R0 ; POINT TO THE OUTPUT BUFFER
204 000416 012702 000000G MOV #G$TBUF,R2 ; POINT AT ARGUMENT BLOCK
205 000422 004767 000000G CALL #EDMSG ; FORMAT THE HEADER
206 000426 PRINT #B$UFF ; PRINT HEADER
207
208 ; PRINT SINK TYPE MESSAGE
209
210 000436 032767 000000G 0C0000G BIT #WC.LOG,$WILD ; ALL LOGGING SINK TYPES?
211 000444 001405 BEQ 50$ ; BR IF NO
212 000446 012704 000322' MOV #SNK$TBL,R4 ; POINT TO SINK TYPE TABLE
213 000452 012403 40$: MOV (R4)+,R3 ; GET LOGGING SINK TYPE
214 000454 001460 BEQ 90$ ; EXIT IF END OF LIST
215 000456 000416 BR 60$ ; CONTINUE
216 000460 012703 0C0001 50$: MOV #FF.CON,R3 ; ASSUME LOGGING CONSOLE
217 000464 032767 000000G 000000G BIT #OM.CON,$OPRND ; LOGGING CONSOLE SPECIFIED?
218 000472 001010 BNE 60$ ; BR IF YES
219 000474 012703 000002 MOV #FF.FIL,R3 ; ASSUME LOGGING FILE
220 000500 032767 000000G 000000G BIT #OM.FIL,$OPRND ; LOGGING FILE SPECIFIED?
221 000506 001002 BNE 60$ ; BR IF YES
222 000510 012703 000004 MOV #FF.MON,R3 ; ELSE MUST BE LOGGING MONITOR
223 000514 004767 000100 60$: CALL PR$TSNK ; PRINT SINK TYPE MESSAGE
224
225 ; PRINT LOGGING STATE
226
227 000520 004767 000204 CALL PR$STA ;
228 000524 112767 000004 177561 MOVB #F.NINF,FLAGS ; ASSUME NO INFORMATION
229 000532 012705 000000G MOV #STMLST,R5 ; GET START OF TEMPLATE LIST
230
231 ; SEARCH FOR THE EVENT TEMPLATE

```

4-	55	MACRO CALLS AND DEFINITIONS
5-	153	LOCAL DATA
6-	264	
6-	265	\$SHXPR - SHOW MODULE X25-PROTOCOL
7-	299	XPRCHA - SHOW X25 PROTOCOL CHARACTERISTICS
8-	343	XPRDTE - SHOW MODULE X25-PROTOCOL DTE
9-	403	PRCHN - PRINT DTE CHANNEL LIST
10-	461	XPRCUG - SHOW MODULE X25-PROTOCOL GROUP
11-	512	\$SHXAC - SHOW MODULE X25-ACCESS
12-	540	
12-	541	\$SHXSV - SHOW MODULE X25-SERVER
13-	574	XSVCHA - SHOW X25 SERVER CHARACTERISTICS
14-	601	
14-	602	\$SHX29 - SHOW MODULE X29-SERVER
15-	635	X29CHA - SHOW X29 SERVER CHARACTERISTICS
16-	657	XACCHA - SHOW X25-ACCESS MODULE CHARACTERISTICS
17-	672	SHOXAD - SHOW X25-ACCESS MODULE DESTINATIONS
18-	701	
18-	705	DISPAT - DISPATCH TO SHOW ROUTINE
19-	742	PRINT - PRINT FORMATTED INFORMATION
20-	775	FMTLIN - FORMAT A LINE ID
21-	812	SHODST - SHOW DESTINATION LIST
22-	900	MOVOPT - SET UP OPTIONAL PARAMETERS

```

540 .SBTTL
541 .SBTTL $SHXSV - SHOW MODULE X25-SERVER
542
543 :+
544 :*** - $SHXSV - SHOW MODULE X25-SERVER
545
546 INPUT:
547 $OPRND = TYPE OF REQUEST
548
549 OUTPUT:
550 THE HEADER IS PRINTED AND THE SHO INFORMATION ROUTINE
551 IS DISPATCHED TO
552
553 $SHXSV::
554 003276 012705 003346' MOV #XSVDTB,R5 ; POINT TO DISPATCH TABLE
555 003302 012704 000000G MOV #SHDRBK,R4 ; POINT TO ARGUMENT BLOCK
556 003306 012724 000012' MOV #XSVL,(R4)+ ; MODULE IS X25-SERVER
557 003312 012724 001662' MOV #XSV,(R4)+ ;
558 003316 016703 000000G MOV $OPRND,R3 ; GET REQUEST TYPE
559 003322 042703 000000C BIC #C<XS.DST!XS.KND>,R3 ; CLEAR DONT CARE BITS
560 003326 012767 177777 174456 MOV #-1,OPTIONS ; WANT ALL OUTPUT
561 003334 012767 115443 174446 MOV #RX25,DTYPE ; X25 DESTINATIONS ONLY
562 003342 CALLR DISPATCH ; DISPATCH TO ROUTINE
563
564 : DISPATCH TABLE
565
566
567 003346 003400' 000000 000017 XSVDTB: .NLIST BEX
568 003356 004252' 000000G 000017 .WORD XSVCHA, 0, CHARL, CHAR ; CHARACTERISTICS
569 003366 004252' 000000G 000022 .WORD SHODST, XS.DST, CHARL, CHAR ; DESTINATION
570 003376 000000 .WORD SHODST, XS.KND, KDSTL, KDST ; KNOWN DESTINATIONS
571 .WORD 0
572 .LIST BEX

```


SHOMOD - LIST MODULE CHARACTER! MACRO V05.03b Saturday 29-Jun-85 05:26 Page 22-2
Symbol table

Size of core pool: 14440 Words (55 Pages)
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:27.02

SY:SHOMOD.V2,[132,134]SHOMOD/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]SHOMOD

SSHPRD - SHOW PROCESS CHARACTERISTICS

```

168 000526 012700 000000G      MOV    # $BUFF,R0      ; POINT AT OUTPUT BUFFER
169 000532 012701 000160'      MOV    # 02MSG,R1     ; POINT AT FORMAT STRING
170 000536 004767 000000G      CALL   $EDMSG         ; FORMAT THE STRING
171 000542          000000G      PRINT  # $BUFF         ; PRINT THE STRING
172 000552 000677          10$   BR      10$           ; REPEAT
173
174          ; DDM/DLC LINE TABLE COUNT
175
176 000554 156512 000012      50$: BISB   C,NLTD(R5),(R2) ; GET LINE TABLE COUNT
177 000560 005062 000002      CLR      2(R2)           ; GET CONTROLLER TABLE COUNT
178 000564 156562 000013 000002 BISB   C,NCTD(R5),2(R2) ;
179 000572 012700 000000G      MOV    # $BUFF,R0      ; POINT AT OUTPUT BUFFER
180 000576 012701 000205'      MOV    # 03MSG,R1     ; POINT AT FORMAT STRING
181 000602 004767 000000G      CALL   $EDMSG         ; FORMAT THE MESSAGE
182 000606          000000G      PRINT  # $BUFF         ; PRINT THE MESSAGE
183 000616 000655          10$   BR      10$           ; AND LOOK FOR NEXT
184
185 000620 000207      60$:  RETURN
186
187
188          000001          .END

```

```

344
345
346
347
348
349
350
351
352
353
354
355
356
357 001102 000000' 000033' 000075' HTBL: .WORD HEADR,HEADR1,HEADR2,HEADR3
358
359 001112
360 001112 012701 001102' PRIHDR: MOV #HTBL,R1 ; point to table of header strings
361 001116 126727 000000G 000000G CMPB $INFO,#IN$CHA ; displaying characteristics ?
362 001124 001001 BNE 10$ ; if ne, no
363 001126 022121 CMP (R1)+,(R1)+ ; else skip to summary header strings
364 001130 005267 177176 10$: INC HFLAG ; ELSE, INDICATE HEADER IS ALREADY PRINTED
365 001134 032767 000000G 000000G BIT #WC.NOD,$WILD ; KNOWN NODES ?
366 001142 001401 BEQ 25$ ; IF EQ, NO ... OKAY
367 001144 005721 TST (R1)+ ; ELSE, DIFFERENT HEADER
368 001146 012700 000000G 25$: MOVL #BUFF,R0 ; POINT TO OUTPUT BUFFER
369 001152 011101 MOV (R1),R1 ; POINT TO THE HEADER STRING
370 001154 012702 000000G MOV #GTBUF,R2 ; POINT AT ARGUMENT BLOCK
371 001160 004767 000000G CALL $EDMSG ; FORMAT THE HEADER
372 001164 PRINT #BUFF ; PRINT HEADER
373 001174 126727 000000G 000000G CMPB $INFO,#IN$CHA ; displaying characteristics ?
374 001202 001404 BEQ 30$ ; if EQ, yes
375 001204 PRINT #FORMAT ; PRINT THE FORMAT HEADER
376 001214 000207 30$: RETURN
377

```

FILEID**SHOSYS

```

SSSSSSSS HH HH 000000 SSSSSSS YY YY SSSSSSS
SSSSSSSS HH HH 000000 SSSSSSS YY YY SSSSSSS
SS HH HH 00 00 SS YY YY SS
SS HH HH 00 00 SS YY YY SS
SS HH HH 00 00 SS YY YY SS
SS HH HH 00 00 SS YY YY SS
SSSSSS HHHHHHHHHH 00 00 SSSSSS YY SSSSSS
SSSSSS HHHHHHHHHH 00 00 SSSSSS YY SSSSSS
SS HH HH 00 00 SS YY SS
SS HH HH 00 00 SS YY SS
SS HH HH 00 00 SS YY SS
SSSSSS HH HH 000000 SSSSSSS YY SSSSSSS
SSSSSS HH HH 000000 SSSSSSS YY SSSSSSS

```

```

LL SSSSSSS iTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSS TT
LL SSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLL SSSSSSS TT
LLLLLLLLL SSSSSSS TT

```

```

56      .SBTTL  MACRO DEFINITIONS
57
58      ; LOCAL MACROS
59
60      .MACRO PRINT TEXT
61      MOV     TEXT,RO      ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
62      CALL    $PRINT      ; PRINT MESSAGE
63      .ENDM  PRINT
64
65      .MACRO ERROR$ TEXT
66
67      ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
68      ; MESSAGE STRING.
69
70      .IF DIF <TEXT><RO>
71      MOV     TEXT,RO      ; GET ADDRESS OF ERROR MESSAGE
72      .ENDC
73      CALL    $CFERR      ; PRINT ERROR MESSAGE
74      .ENDM  ERROR$
75
76      .MACRO EPRINT TEXT
77
78      ; PRINT TEXT ON ERROR LUN
79
80      .IF DIF <TEXT><RO>
81      MOV     TEXT,RO      ; GET ADDRESS OF ASCIZ STRING TO PRINT
82      .ENDC
83      CALL    $CFER1      ; PRINT MESSAGE ON ERROR LUN
84      .ENDM  EPRINT
85
86
87      ; MACRO CALLS
88
89      .MCALL  ISTAT$,STATE$,TRANS
90

```

TMPBUF CREATED BY MACRO ON 29-JUN-85 AT 05:28 PAGE 1 D 10
 SYMBOL CROSS REFERENCE CREF 04.00

SYMBOL		VALUE	REFERENCES								
BUF	KT	000000 RG	#6-140	7-212							
BUF	ST	000000 RG	#6-140	7-213							
CS	BUF	= ***** GX	7-210								
C	CBNM	000004	#5-104	*8-256							
C	BSZ	000006	#5-105	*9-287							
C	LEN	= 000022	#5-111	7-207							
C	RBNM	000010	#5-106	7-223	*10-317						
C	RBSZ	000012	#5-107	7-224	*11-350						
C	SBNM	000014	#5-108	7-226	*12-381						
C	SBSZ	000016	#5-109	7-227	*13-410						
C	STS	= ***** GX	*7-210								
C	THSH	000020	#5-110	*14-440							
HIC	BNM	000004 R	#5-122	8-260							
HIC	BSZ	000010 R	#5-124	9-290							
HIR	BNM	000014 R	#5-126	10-323							
HIR	BSZ	000020 R	#5-128	11-354							
HIS	BSZ	000026 R	#5-131	13-413							
HITHSH		000032 R	#5-133	*10-319	*10-320	14-444					
LOC	BNM	000002 R	#5-121	8-258							
LOC	BSZ	000006 R	#5-123	9-288							
LOR	BNM	000012 R	#5-125	10-321							
LOR	BSZ	000016 R	#5-127	11-352							
LOS	BNM	000022 R	#5-129	12-383							
LOS	BSZ	000024 R	#5-130	13-411							
LOTHSH		000030 R	#5-132	14-442							
SET	CBN	000134 R	#8-253								
SET	CBS	000173 R	#9-284								
SET	CBN	000234 R	#10-314								
SET	CBN	000306 R	#11-347								
SET	CBN	000350 R	#12-378								
SET	CBN	000404 R	#13-407								
SET	THR	000442 R	#14-437								
TEMP		000000 R	#5-114	*7-209	7-220	8-254	9-285	10-315	11-348	12-379	13-408
			14-438								
\$ALPHA	=	000022	#6-140								
\$ANY	=	000020	#6-140								
\$BLANK	=	000006	#6-140								
\$CCBNM	=	***** GX	*8-257								
\$DIGIT	=	000024	#6-140								
\$DNUMB	=	000014	#6-140								
\$EOS	=	000012	#6-140								
\$ERROR	=	***** GX	*7-216								
\$EXIT	=	000000	#6-140								
\$FAIL	=	17 7	#6-140								
\$GPRM	=	* ***	6-140								
\$LAMDA	=	000000	#6-140								
\$LIMITB	=	000002 RG	#5-119								
\$MINSB	=	***** GX	5-129								
\$NUMBR	=	000002	#6-140								
\$RAD50	=	000016	#6-140								
\$RDBNM	=	***** GX	*10-318								
\$RDBSZ	=	***** GX	*11-351								

TMPCHN CREATED BY MACRO ON 29-JUN-85 AT 05:29 PAGE 1 D 11

SYMBOL CROSS REFERENCE CREF 04.00

SYMBOL	VALUE	REFERENCES
CHNKTB	000000 RG	#6-141 7-189
CHNSTB	000000 RG	#6-141 7-190
CH\$HI	= ***** GX	8-216
CS.CHN	= ***** GX	7-186
C\$HHI	= ***** GX	8-216
C\$HLEN	= ***** GX	7-181
C\$HLO	= ***** GX	8-214
ERRCHN	= 000000 R	#5-132
EX\$SEV	= 000004	#5-131 5-132
SETHI	000164 R	#8-216
SETLC	000144 R	#8-214
\$ALPHA	= 000022	#6-141
\$ANY	= 000020	#6-141
\$BLANK	= 000006	#6-141
\$CFERR	= ***** GX	7-195
\$DIGIT	= 000024	#6-141
\$DNUMB	= 000014	#6-141
\$EOS	= 000012	#6-141
\$ERROR	= ***** GX	*7-196 *7-197
\$EXIT	= 000000	#6-141
\$FAIL	= 177777	#6-141
\$GPRM	= *****	6-141
\$GTNUM	= ***** GX	8-214 8-216
\$LAMPD	= 000000	#6-141
\$NUMBR	= 000002	#6-141
\$RAD50	= 000016	#3-141
\$RONLY	= *****	6-141 6-141 6-141
\$STRNG	= 000004	#6-141
\$SUBXP	= 000010	#6-141
\$TALOC	= ***** GX	7-182
\$TCHN	= 000040 RG	#7-179
\$SPLG	= 177777	#6-141
\$SKEY	= 177777	#6-141
.ERROR	= ***** GX	*7-187 7-193
.FLAGS	= ***** GX	*7-185 8-216
.TEMP	= ***** GX	*7-183
.TPARS	= ***** GX	7-191

TMPCNT - PARSE CONTROLLER DEFIN MACRO V05.03b Saturday 29-Jun-85 05:29 Page 11-1

Symbol table

CNTKTB 000000RG	003 C.URM 000014	TEMP 000000R	\$EXIT = 000000	\$TMCNT 000002RG
CNTSTB 000000RG	002 C.VEC 000006	\$ALPHA= 000022	\$FAIL = 177777	\$\$\$FLG= 177777
CS.CNT= ***** GX	RCDEV 000040R	002 \$ANY = 000020	\$LAMBDA= 000070	\$\$\$KEY= 177777
C.CNT 000004	SETCNT 000062R	\$BLANK= 000006	\$NUMBR= 000012	\$\$\$STA= 000000
C.CSR 000010	SETCSR 000164R	\$DIGIT= 000024	\$RAD50= 000016	.PNUMB= ***** GX
C.LEN = 000016	SETPRI 000222R	\$DNUMB= 000014	\$STRNG= 000004	.PNUMH= ***** GX
C.PRI 000012	SETURM 000262R	\$EOS = 000012	\$SUBXP= 000010	.TPARS= ***** GX
C.STS = ***** GX	SETVEC 000116R	\$ERROR= ***** GX	\$TALOC= ***** GX	

. ABS. 000016 000 (RW,I,GBL,ABS,OVR)
 000310 001 (RW,I,LCL,REL,CON)
 \$STATE 000052 002 (RW,D,LCL,REL,CON)
 \$KTAB 000000 003 (RW,D,LCL,REL,CON)
 \$KSTR 000000 004 (RW,D,LCL,REL,CON)
 Errors detected: 0

*** Assembler statistics

Work file reads: 0
 Work file writes: 0
 Size of work file: 10406 Words (41 Pages)
 Size of core pool: 14440 Words (55 Pages)
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:22.34
 SY:TMPCNT.V2,[132,134]TMPCNT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPCNT

TMPUCUC CREATED BY MACRO ON 29-JUN-85 AT 05:29 PAGE 1 D 13
 SYMBOL CROSS REFERENCE CREF 04.00

SYMBOL	VALUE	REFERENCES
CS.CUG	= ***** GX	7-214
CS.FLG	= ***** GX	7-247
CUGKTB	000000 RG	#6-158 7-218
CUGSTB	000000 RG	#6-158 7-219
C.CUG	000010	#5-137 7-245
C.FLG	000012	#5-138 7-247
C.LEN	= 000013	#5-139 7-209
C.NAM	000004	#5-136 7-243
ERRCUG	000031 R	#5-148
ERRFLG	000066 R	#5-149
ERRNAM	000000 R	#5-147
EX\$SEV	= 000004	#5-146 5-147 5-148 5-149
G\$LENX	= ***** GX	7-216
SETCUG	000244 R	#7-245
SETFLG	000264 R	#7-247
SETNAM	000230 R	#7-243
\$ALPHA	= 000022	#6-158
\$ANY	= 000020	#6-158
\$BLANK	= 000006	#6-158
\$BYTMN	= ***** GX	*7-216
\$CFERR	= ***** GX	7-224
\$DIGIT	= 000024	#6-158
\$DNUMB	= 000014	#6-158
\$EOS	= 000012	#6-158
\$ERROR	= ***** GX	*7-225 *7-226
\$EXIT	= 000000	#6-158
\$FAIL	= 177777	#6-158
\$GPRM	= *****	6-158
\$GTNMB	= ***** GX	7-247
\$GTNUM	= ***** GX	7-245
\$GTR50	= ***** GX	7-243
\$LAMDA	= 000000	#6-158
\$NUMBR	= 000002	#6-158
\$RAD50	= 000016	#6-158
\$RONLY	= *****	6-158 6-158
\$STRNG	= 000004	#6-158
\$SUBXP	= 000010	#6-158
\$TALOC	= ***** GX	7-210
\$TMCUG	000116 RG	#7-207
\$SFLG	= 177777	#6-158
\$SKEY	= 177777	#6-158
.ERROR	= ***** GX	*7-215 7-222
.FLAGS	= ***** GX	*7-213 7-247
.TEMP	= ***** GX	*7-211
.TPARS	= ***** GX	7-220

```

277      .SBTTL SETNCT - STORE NUMBER OF CONTROLLER TABLES
278
279      ;*** - SETNCT - STORE NUMBER OF CONTROLLER TABLES
280
281      :INPUTS:
282      :.PNUMB = NUMBER OF CONTROLLER TABLES TO BE SET IN PROCESS SPACE
283
284      :OUTPUTS:
285      :NUMBER IS VALIDATED AND THE TRANSITION IS ACCEPTED/REJECTED
286
287      :-
288
289      SETNCT:
290      000230      MOV      TEMP,RO      ; GET THE TEMPLATE ADDRESS
291      000230      016700      177546      MOV      .PNUMB,C.NCT(R0)      ; STORE THE CONTROLLER COUNT
292      000234      116760      000000G 000013      CMVB     .PNUMB,C.NCT(R0)      ; STORE THE CONTROLLER COUNT
293      000242      026727      000000G 000000G      CMP      .PNUMB,#$MXNCT      ; VALUE IN RANGE ?
294      000250      101402      BLOS     10$      ; IF LOS, YES
295      000252      062716      000002      ADD      #2,(SP)      ; ELSE, REJECT
296      000256      000207      10$:      RETURN
297
298      000001      .END
  
```

```

222      .SDTTL SETINC - INCOMING TIMER
223
224      *** - SETINC - SETUP INCOMING TIMER
225
226      THIS ACTION ROUTINE SETS UP THE INCOMING TIMER VALUE FOR THE EXECUTOR
227
228      INPUT:
229      .PNUMB = INCOMING TIMER
230      .PNUMH = (HIGH ORDER)
231
232      OUTPUT:
233      THE INCOMING TIMER IS STORED IN C.INCT OF THE TEMPLATE
234
235      -
236
237      SETINC:
238      000156      016700      177616      MOV      TEMP,R0      ; GET START OF TEMPLATE
239      000162      116760      000000G 000012      MOVB     ,PNUMB,C.INCT(R0) ; STORE THE OUTGOING TIMER
240      000170      001406      BEQ      10$      ; BR IF INVALID VALUE
241      000172      005767      000000C      TST      ,PNUMH      ; IS IT ONLY A WORD VALUE ?
242      000176      001003      BNE      10$      ; BR IF NO, ERROR
243      000200      105767      000001G      TSTB     ,PNUMB+1      ; IS IT A BYTE VALUE ?
244      000204      001402      BEQ      20$      ; IF EQ, YES - OKAY
245      000206      062716      000002      ADD      #2,(SP)      ; ELSE, REJECT TRANSITION
246      000212      000207      10$:      RETURN
247

```

[illegible]

```

LL          SSSSSSSS  ?TTTTTTTT
LL          SSSSSSSS  T'TTTTTTT
          SS          TT
          SS          TT
          SS          TT
          SS          TT
          SSSSSS      TT
          SSSSSS      TT
          SS          TT
          SS          TT
          SS          TT
          SS          TT
          SS          TT
          SSSSSSSS  TT
          SSSSSSSS  TT
          SSSSSSSS  TT

```

```

663          .SBTTL FMTLIN - FORMAT A LINE ID
664          :+
665          *** - FMTLIN - FORMAT A LINE ID
666          :
667          INPUTS:
668          $FLAGS = FLAGS WORD
669          :
670          OUTPUTS:
671          LINE ID FORMATTED INTO OUTPUT BUFFER
672          :
673          :-
674
675 003426 012700 000226' FMTLIN: MOV #LINID,R0 ; GET ADDRESS OF OUTPUT BUFFER
676 003432 012703 000000G FMTLN2: MOV #SDVNAM,R3 ; POINT AT LINE DESCRIPTOR
677 003436 016301 000000G MOV L$NAM(R3),R1 ; GET DEVICE NAME
678 003442 004767 000000G CALL $C5TA ; CONVERT NAME TO ASCII
679 003446 112720 000055 MOVB #'-(R0)+ ; INSERT DELIMITER
680 003452 116301 000000G MOV L$CNT(R3),R1 ; GET CONTROLLER NUMBER
681 003456 012702 014012 MOV #<10.!<3*4000>>,R2 ; BASE=10., WIDTH = 3
682 003462 004767 000000G CALL $CBTA ; CONVERT TO ASCII
683 003466 032767 000000G 000000G BIT #FL.MUX,$FLAGS ; IS THIS A MUX DEVICE ?
684 003474 001410 BEQ 10$ ; IF EQ, NO
685 003476 112720 000055 MOVB #'-(R0)+ ; INSERT DELIMITER
686 003502 116301 000000G MOV L$UNT(R3),R1 ; ELSE, GET UNIT NUMBER
687 003506 012702 014012 MOV #<10.!<3*4000>>,R2 ; BASE=10., WIDTH = 3
688 003512 004767 000000G CALL $CBTA ; CONVERT TO ASCII
689 003516 032767 000000G 000000G 10$: BIT #FL.MTP,$FLAGS ; IS THE LINE MULTI-POINT ?
690 003524 001414 BEQ 20$ ; IF EQ, NO
691 003526 032767 000000G 000000G BIT #FL.TRB,$FLAGS ; DID THE USER WANT TRIBUTARY ?
692 003534 001410 BEQ 20$ ; IF EQ, NO
693 003536 112720 000056 MOVB #'-(R0)+ ; INSERT DELIMITER
694 003542 016301 000000G MOV L$TRI(R3),R1 ; ELSE, GET TRIBUTARY NUMBER
695 003546 012702 014012 MOV #<10.!<3*4000>>,R2 ; BASE=10., WIDTH = 3
696 003552 004767 000000G CALL $CBTA ; CONVERT TO ASCII
697 003556 105010 20$: CLRB (R0) ; MAKE STRING ASCII
698 003560 000207 RETURN
699
700          000001 .END
701

```

```

232
233 000536 011505          70$: MOV (R5),R5          ; GET NEXT TEMPLATE
234 000540 001412          BEQ 80$                ; IF EQ, END OF TEMPLATES
235 000542 122765 000000G 000000G CMPB #CS.EVT,C.STS(R5) ; IS THIS AN EVENT TEMPLATE?
236 000550 001372          BNE 70$                ; IF NE, NO .. KEEP LOOKING
237 000552 020365 000016    CMP R3,C.SNK(R5)       ; IS THIS FOR THE CORRECT SINK?
238 000556 001367          BNE 70$                ; IF NE, NO .. KEEP LOOKING
239
240          ; PRINT THE EVENT LOGGING CHARACTERISTICS
241
242 000560 004767 000230    CALL PRTEVT           ; PRINT EVENT MESSAGE
243 000564 000764          BR 70$                 ; LOOK FOR NEXT EVENT TEMPLATE
244 000566 132767 000004 177517 80$: BITB #E.NINF,FLAGS ; ANY EVENT INFORMATION DISPLAYED?
245 000574 001404          BEQ 85$                 ; BR IF YES
246 000576          PRINT #NOINFO                 ; DISPLAY NO INFORMATION MESSAGE
247 000606 032767 000000G 000000G 85$: BIT #WC.LOG,$WILD ; KNOWN LOGGING SPECIFIED?
248 000614 001316          BNE 40$                 ; BR IF YES
249
250 000616 000207          90$: RETURN              ; RETURN TO CALLER
251

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53

```
.TITLE SHOMOD - LIST MODULE CHARACTERISTICS
.IDENT /V05.00/
.ENABL LC
```

```
: COPYRIGHT (C) 1981, 1982, 1983, 1985 BY
: DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
```

```
: THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
: ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
: INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
: COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
: OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
: TRANSFERRED.
```

```
: THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
: AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
: CORPORATION.
```

```
: DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
: SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
```

```
: MODULE DESCRIPTION:
```

```
: CFE - SHOW MODULE CHARACTERISTICS
```

```
: DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING
```

```
: IDENT HISTORY:
```

- ```
: 1.00 29-OCT-81
: DECNET-11M/S V3.1
: DECNET-11M-PLUS V1.1
:
: 3.00 16-APR-82
: DECNET-11M V3.1
: DECNET-11M-PLUS V1.1
:
: 4.00 07-NOV-83
: DECNET-11M V4.0
: DECNET-11M-PLUS V2.0
:
: 5.00 22-JUL-85
: DECnet-11M/S V4.2
: DECnet-11M-Plus V3.0
: DECnet-Micro/RSX V1.0
```

```

574 .SBTTL XSVCHA - SHOW X25 SERVER CHARACTERISTICS
575 ;+
576 ** XSVCHA - SHOW X25 SERVER CHARACTERISTICS
577
578 INPUT $OPRND = REQUEST TYPE
579
580 OUTPUT THE X25 SERVER CHARACTERISTICS ARE DISPLAYED.
581 ;+
582
583 XSVCHA:
584 003400 MOV #XSVCBK,R1 ; POINT TO ARGUMENT BLOCK
585 003400 C12701 000000' MOV $PSIPT,R0 ; POINT TO PSN$DF
586 003400 016700 000000G MOV $PSPOR(R0),(R1) ; MAXIMUM CIRCUITS
587 003410 016011 000000G DEC (R1)+ ; MAKE FUDGE FACTOR INVISIBLE TO USER
588 003414 005321 MOV #STMLST,R0 ; POINT TO LISTHEAD
589 003416 012700 000000G 10$: MOV (R0),R0 ; NEXT TEMPLATE
590 003422 011000 BEQ 20$; IF EQ, NO INFORMATION
591 003424 001416 CMPB #CS.LLC,C.STS(R0) ; IS THIS AN LLC DESCRIPTOR ?
592 003426 122760 000000G 000000G BNE 10$; BR IF NO
593 003434 001372 CMP L$LNAM(R0),#<^RNW> ; IS THIS LLC FOR SERVER MODULE ?
594 003436 026027 000000G 055430 BNE 10$; BR IF NO
595 003444 001366 MOV L$LCM(R0),(P1)+ ; COUNTER TIMER
596 003446 016021 000000G MOV #XSVCFTR,R3 ; POINT TO FORMATTING TABLE
597 003452 012703 001260' CALL PRINT ; SHOW INFORMATION
598 003456 004767 000350 20$: RETURN
599 003462 000207

```



SHOMOD CREATED BY MACRO ON 29-JUN-85 AT 05:27 PAGE 1 E 5  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL  | VALUE      | REFERENCES                                                                |
|---------|------------|---------------------------------------------------------------------------|
| BLANK   | 002022 R   | #5-251 5-251 10-501                                                       |
| BLANKL  | = 000001   | #5-251 10-500                                                             |
| CHAR    | 001720 R   | #5-246 5-246 6-290 6-291 6-293 11-534 11-535 12-567 12-568                |
| CHARL   | = 000017   | #5-146 14-628 14-629 6-290 6-291 6-293 11-534 11-535 12-567 12-568 14-628 |
| CHNBLK  | 001101 R   | #5-205 9-451                                                              |
| CHNBND  | = 000130 R | #5-167 9-440                                                              |
| CHNBUF  | 000034 R   | #5-166 9-417 9-429 9-449 9-450 9-455 9-457 21-863 21-864                  |
| CHNHDR  | 001064 R   | #5-218 21-875 9-416                                                       |
| CH\$HI  | = ***** GX | 9-424                                                                     |
| CS.CHN  | = ***** GX | 9-421 9-444                                                               |
| CS.CUG  | = ***** GX | 10-483                                                                    |
| CS.DST  | = ***** GX | 21-830                                                                    |
| CS.DTE  | = ***** GX | 8-358 10-476 10-481                                                       |
| CS.LLC  | = ***** GX | 13-591                                                                    |
| CS.MSK  | = ***** GX | 21-888                                                                    |
| CS.NUM  | = ***** GX | 21-880                                                                    |
| CS.RDT  | = ***** GX | 17-685                                                                    |
| CS.SHI  | = ***** GX | 21-883                                                                    |
| CS.SLO  | = ***** GX | 21-882                                                                    |
| CS.UNI  | = ***** GX | 8-390                                                                     |
| CS.VAL  | = ***** GX | 21-889                                                                    |
| CS.X29  | = ***** GX | 15-648                                                                    |
| CS.X3P  | = ***** GX | 7-312                                                                     |
| CUGBK   | 000000 R   | #5-212 10-491                                                             |
| CUGB1   | 000000 R   | #5-212 5-212                                                              |
| CUGB2   | 000004 R   | #5-212 5-212                                                              |
| CUGFT   | 001170 R   | #5-212 10-506                                                             |
| CUGM1   | 001116 R   | #5-210 5-212                                                              |
| CUGM2   | 001134 R   | #5-211 5-212                                                              |
| CUSBIL  | = ***** GX | 10-502                                                                    |
| C\$HLO  | = ***** GX | 9-438                                                                     |
| C\$UCUG | = ***** GX | 10-494                                                                    |
| C\$UFLG | = ***** GX | 10-502                                                                    |
| C\$UNAM | = ***** GX | 10-487 10-489 10-492 10-493                                               |
| C.STS   | = ***** GX | 7-312 8-358 8-390 9-421 9-424 9-444 10-476 10-481 10-483                  |
|         |            | 13-591 15-648 17-685 21-830 22-925                                        |
| DA\$CUG | = ***** GX | 21-881                                                                    |
| DISPAT  | 003750 R   | 6-285 11-529 12-562 14-623 #18-721                                        |
| DSTBK   | 000000 R   | #5-231 21-858                                                             |
| DSTB1   | 000000 R   | #5-231 5-231                                                              |
| DSTB10  | 000062 R   | #5-231 5-231                                                              |
| DSTB2   | 000004 R   | #5-231 5-231                                                              |
| DSTB3   | 000012 R   | #5-231 5-231                                                              |
| DSTB4   | 000024 R   | #5-231 5-231                                                              |
| DSTB5   | 000030 R   | #5-231 5-231                                                              |
| DSTB6   | 000032 R   | #5-231 5-231                                                              |
| DSTB7   | 000034 R   | #5-231 5-231                                                              |
| DSTFT   | 001532 R   | #5-231 21-893                                                             |
| DSTM1   | 001266 R   | #5-223 5-231                                                              |

SHOPRO - LIST PROCESS CHARACTER MACRO V05.03b Saturday 29-Jun-85 05:27 <sup>E 6</sup> Page 5-2  
Symbol table

|                  |                |                  |                   |                   |
|------------------|----------------|------------------|-------------------|-------------------|
| CLEAR 000266R    | C.FLG 000006   | C.STS = ***** GX | O2MSG 000160R     | \$FNPRO= ***** GX |
| CLEARL= 000007   | C.LIN 000011   | HEADR 000000R    | O3MSG 000205R     | \$GTBUF= ***** GX |
| CL.LEN= 000020   | C.NAM 000004   | HEADR1 000046R   | PROMSG 000124R    | \$PRINT= ***** GX |
| CS.LLC= ***** GX | C.NCTD= 000013 | MSGBLK 000276R   | WC.NAM= ***** GX  | \$SHPRO 000302RG  |
| C.CTM 000016     | C.NLT 000014   | ON 000264R       | ZF.MFL= ***** GX  | \$TMLST= ***** GX |
| C.DEV 000012     | C.NLTD= 000012 | ONL = 000002     | \$BUFF = ***** GX | \$WILD = ***** GX |
| C.EXT 000015     | C.PRI 000010   | O1MSG 000142R    | \$EDMSG= ***** GX |                   |

. ABS. 000020 000 (RW,I,GBL,ABS,OVR)  
000622 001 (RW,I,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 8865 Words ( 35 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:06.19  
SY:SHOPRO.V2,[132,134]SHOPRO/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]SHOPRO

```

379
380
381
382
383
384
385
386
387
388
389
390 001216
391 001216 010146
392 001220 000241
393 001222 006201
394 001224 006201
395 001226 000301
396 001230 042701 177700
397 001234 010122
398 001236 042716 176000
399 001242 012622
400 001244 000207

```

```

.SBTTL PUTADR - PUT A PHASE 4 ADDRESS
;+
; ** PUTADR - PUT A PHASE 4 NODE ADDRESS
; INPUTS: R1 = NODE ADDRESS
; R2 = ARG LIST POINTER
; OUTPUT: R1 DESTROYED.
; R2 UPDATED, TWO WORD ADDRESS PUT IN ARG LIST.
;-
PUTADR:
MOV R1,-(SP) ; SAVE ADDRESS
CLC ; WANT HI ORDER ZEROES
ASR R1 ; SHIFT AREA
ASR R1 ; INTO HI BYTE
SWAB R1 ; MAKE THAT LO BYTE
BIC #C77,R1 ; ISOLATE AREA
MOV R1,(R2)+ ; STUFF AREA INTO ARG LIST
BIC #C177,(SP) ; ISOLATE NODE ADDRESS
MOV (SP)+,(R2)+ ; STUFF IT INTO LIST
RETURN

```

SHOSYS - CFE SHOW SYSTEM PARAM MACRO V05.03b Saturday 29-Jun-85 F 8  
Table of contents 05:28

3- 136 \$SHSYS - SHOW SYSTEM CHARACTERISTICS

```

LOCAL DATA
92 .SBTTL LOCAL DATA
93
94 ; LOCAL DATA
95 ;
96 ;
97 ;
98 ; DEFINE THE BUFFER BUFFER BLOCK
99 ;
100 000000 .ASECT
101 000000 .:=0
102 000000 .BLKW 1 ; C.LNK
103 000002 .BLKW 1 ; C.STS
104 000004 C.CBNM: .BLKW 1 ; CCB NUMBER
105 000006 C.CBSZ: .BLKW 1 ; CCB SIZE
106 000010 C.RBNM: .BLKW 1 ; RDB NUMBER
107 000012 C.RBSZ: .BLKW 1 ; RDB SIZE
108 000014 C.SBNM: .BLKW 1 ; SDB NUMBER
109 000016 C.SBSZ: .BLKW 1 ; SDB NUMBER
110 000020 C.HSH: .BLKW 1 ; RDB THRESHOLD VALUE
111 000022 C.LEN=. ; LENGTH OF THE BLOCK
112 000000 .PSECT
113
114 000000 TEMP: .BLKW 1 ; TEMPLATE ADDRESS
115
116 ;
117 ; DEFINE THE LIMITS TABLE (*** LIMITS SHOULD AGREE WITH THE NETGEN LIMITS ***)
118 ;
119 000002 $LIMITB: ; START OF TABLE LIMITS
120 ;
121 000002 000004 LOCBNM: .WORD 4. ; GLOBAL FOR 'GBLPAT' IF REQUIRED
122 000004 000310 HICBNM: .WORD 200. ; MINIMUM NUMBER OF CCB'S
123 000006 000034 LOCBSZ: .WORD 28. ; MAXIMUM NUMBER OF CCB'S
124 000010 000144 HICBSZ: .WORD 100. ; MINIMUM SIZE OF CCB'S
125 000012 000004 LORBNM: .WORD 4. ; MAXIMUM SIZE OF CCB'S
126 000014 000310 HIRBNM: .WORD 200. ; MINIMUM NUMBER OF RDB'S
127 000016 000100 LORBSZ: .WORD 64. ; MAXIMUM NUMBER OF RDB'S
128 000020 007726 HIRBSZ: .WORD 4054. ; MINIMUM SIZE OF RDB'S
129 000022 000000G LOSBNM: .WORD $MINSB ; MAXIMUM SIZE OF RDB'S
130 000024 000000 LOSBSZ: .WORD 0. ; MINIMUM NUMBER OF SDB'S
131 000026 007726 HISBSZ: .WORD 4054. ; MINIMUM SIZE OF SDB'S
132 000030 000001 LOTHSH: .WORD 1. ; MAXIMUM THRESHOLD VALUE
133 000032 000000 HITHSH: .WORD 0. ; MAXIMUM THRESHOLD VALUE, DEPENDS ON # OF RDB'S

```

TMPBUF      CREATED BY MACRO ON 29-JUN-85 AT 05:28      PAGE 2      E 10  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE       | REFERENCES                                                                               |
|-----------|-------------|------------------------------------------------------------------------------------------|
| \$RONLY   | = *****     | 6-140                                                                                    |
| \$SDBNM   | = ***** GX  | *12-382      6-140                                                                       |
| \$STRNG   | = 000004    | #6-140                                                                                   |
| \$SUBXP   | = 000010    | #6-140                                                                                   |
| \$TALOC   | = ***** GX  | 7-208                                                                                    |
| \$THRSH   | = ***** GX  | *14-441                                                                                  |
| \$TMBUF   | = 000034 RG | #7-205                                                                                   |
| \$\$\$FLG | = 177777    | #6-140                                                                                   |
| \$\$\$KEY | = 177777    | #6-140                                                                                   |
| .PNUMB    | = ***** GX  | 8-255      9-286      10-316      11-349      12-380      13-409      14-439      14-444 |
| .TPARS    | = ***** GX  | 7-214                                                                                    |

TMPCHN      CREATED BY    MACRO    ON 29-JUN-85 AT 05:29      PAGE 2      E 11  
 MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|
| CALLR   | #4-124 |        |        |        |        |        |        |
| DBGTP\$ | #6-141 | #6-151 |        |        |        |        |        |
| EPRINT  | #4-78  |        |        |        |        |        |        |
| ERROR\$ | #4-67  | 7-195  |        |        |        |        |        |
| ISTAT\$ | #4-124 | 6-141  |        |        |        |        |        |
| MTRAN\$ | #6-141 |        |        |        |        |        |        |
| PRINT   | #4-62  |        |        |        |        |        |        |
| STAT\$  | #4-124 | 6-145  | #6-147 | #6-149 | #6-153 | #6-155 | #6-161 |
| TRANS   | #4-124 | #6-146 | #6-148 | #6-150 | #6-151 | #6-154 | #6-156 |
| \$GTNMB | #4-100 |        |        |        |        |        |        |
| \$GTNUM | #4-90  | 8-214  | 8-216  |        |        |        |        |
| \$GTR50 | #4-112 |        |        |        |        |        |        |

IMPCNT CREATED BY MACRO ON 29-JUN-85 AT 05:29 PAGE 1 E 12  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL    | VALUE      | REFERENCES                                        |
|-----------|------------|---------------------------------------------------|
| CNTKTB    | 000000 RG  | #5-95 6-159                                       |
| CNTSTB    | 000000 RG  | #5-95 6-160                                       |
| CS.CNT    | = ***** GX | 6-157                                             |
| C.CNT     | 000004     | #4-82 *7-184                                      |
| C.CSR     | 000010     | #4-84 *9-234                                      |
| C.LEN     | = 000016   | #4-87 6-153                                       |
| C.PRI     | 000012     | #4-85 *10-261                                     |
| C.STS     | = ***** GX | *6-157                                            |
| C.URM     | 000014     | #4-86 *6-156 *11-286                              |
| C.VEC     | 000006     | #4-85 *8-208                                      |
| SETCNT    | 000062 R   | #7-182                                            |
| SETCSR    | 000164 R   | #9-232                                            |
| SETPRI    | 000222 R   | #10-259                                           |
| SETURM    | 000262 R   | #11-284                                           |
| SETVEC    | 000116 R   | #8-206                                            |
| TEMP      | 000000 R   | #4-74 *6-155 7-183 8-207 9-233 10-260 11-285      |
| \$ALPHA   | = 000022   | #5-95                                             |
| \$ANY     | = 000020   | #5-95                                             |
| \$BLANK   | = 000006   | #5-95                                             |
| \$DIGIT   | = 000024   | #5-95                                             |
| \$DNUMB   | = 000014   | #5-95                                             |
| \$EOS     | = 000012   | #5-95                                             |
| \$ERROR   | = ***** GX | *6-163                                            |
| \$EXIT    | = 000000   | #5-95                                             |
| \$FAIL    | = 177777   | #5-95                                             |
| \$GPRM    | = *****    | 5-95                                              |
| \$LAMDA   | = 000000   | #5-95                                             |
| \$NUMBR   | = 000002   | #5-95                                             |
| \$RAD50   | = 000016   | #5-95                                             |
| \$RONLY   | = *****    | 5-95 5-95                                         |
| \$STRNG   | = 000004   | #5-95                                             |
| \$SUBXP   | = 000010   | #5-95                                             |
| \$TALOC   | = ***** GX | 6-154                                             |
| \$TMCNT   | 000002 RG  | #6-151                                            |
| \$\$\$FLG | = 177777   | #5-95                                             |
| \$\$\$KEY | = 177777   | #5-95                                             |
| .PNUMB    | = ***** GX | 7-184 8-208 8-211 8-213 9-234 9-237 10-261 10-262 |
| .PNUMH    | = ***** GX | 10-264 11-286                                     |
| .TPARS    | = ***** GX | 7-185 8-209 9-235 11-287                          |
|           |            | 6-161                                             |



TMPUG      CREATED BY    MACRO    ON 29-JUN-85 AT 05:29      PAGE 2      E 13  
 MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |  |  |  |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--|--|--|
| CALLR   | #4-124 |        |        |        |        |        |        |        |        |        |  |  |  |
| DBGTP\$ | #6-158 | #6-175 | #6-178 |        |        |        |        |        |        |        |  |  |  |
| EPRINT  | #4-78  |        |        |        |        |        |        |        |        |        |  |  |  |
| ERROR\$ | #4-67  | 7-224  |        |        |        |        |        |        |        |        |  |  |  |
| ISTAT\$ | #4-124 | 6-158  |        |        |        |        |        |        |        |        |  |  |  |
| MTRAN\$ | #6-158 |        |        |        |        |        |        |        |        |        |  |  |  |
| PRINT   | #4-62  |        |        |        |        |        |        |        |        |        |  |  |  |
| STAT\$  | #4-124 | 6-162  | #6-164 | #6-166 | #6-169 | #6-171 | #6-173 | #6-176 | #6-180 | #6-182 |  |  |  |
|         | #6-184 | #6-189 |        |        |        |        |        |        |        |        |  |  |  |
| TRAN\$  | #4-124 | #6-163 | #6-165 | #6-167 | #6-170 | #6-172 | #6-174 | #6-175 | #6-177 | #6-178 |  |  |  |
|         | #6-181 | #6-183 | #6-185 |        |        |        |        |        |        |        |  |  |  |
| \$GINMB | #4-100 | 7-247  |        |        |        |        |        |        |        |        |  |  |  |
| \$GTNUM | #4-90  | 7-245  |        |        |        |        |        |        |        |        |  |  |  |
| \$GTR50 | #4-112 | 7-243  |        |        |        |        |        |        |        |        |  |  |  |

TMPDDM - PARSE DEVICE PROCESS D MACRO V05.03b Saturday 29-Jun-85 05:30 Page 11-1

## Symbol table

|         |          |     |         |         |     |          |        |    |          |        |    |            |          |     |
|---------|----------|-----|---------|---------|-----|----------|--------|----|----------|--------|----|------------|----------|-----|
| BITS    | 000020R  | 002 | NCT     | 000212R | 002 | ZF.DLC=  | *****  | GX | \$CAT5 = | *****  | GX | \$STRNG=   | 000004   |     |
| BITO    | 000032R  | 002 | PRIOR   | 000166R | 002 | ZF.DVP=  | *****  | GX | \$DIGIT= | 000024 |    | \$SUBXP=   | 000010   |     |
| CS.DDM= | *****    | GX  | SETFLG  | 000124R |     | ZF.KMX=  | *****  | GX | \$DNUMB= | 000014 |    | \$TALOC=   | *****    | GX  |
| C.FLG   | 000006   |     | SETNAM  | 000064R |     | ZF.LMC=  | *****  | GX | \$EOS =  | 000012 |    | \$TMDDM    | 000004RG |     |
| C.LEN = | 000014   |     | SETNCT  | 000230R |     | ZF.MAN=  | *****  | GX | \$ERROR= | *****  | GX | \$\$\$FLG= | 177777   |     |
| C.NAM   | 000004   |     | SETNLT  | 000200R |     | ZF.MFL=  | *****  | GX | \$EXIT = | 000000 |    | \$\$\$KEY= | 000014   |     |
| C.NCT   | 000013   |     | SETPRI  | 000140R |     | ZF.MUX=  | *****  | GX | \$FAIL = | 177777 |    | \$\$\$STA= | 000030   |     |
| C.NLT   | 000012   |     | TEMP    | 000002R |     | ZF.PSE=  | *****  | GX | \$LAMDA= | 000000 |    | \$\$\$TMP= | 000060R  | 004 |
| C.PRI   | 000010   |     | ZF      | 000226R | 002 | ZF.TIM=  | *****  | GX | \$MXNCT= | *****  | GX | .PNUMB=    | *****    | GX  |
| C.STS = | *****    | GX  | ZF.COU= | *****   | GX  | \$ALPHA= | 000022 |    | \$MXNLT= | *****  | GX | .PSTCN=    | *****    | GX  |
| DDMKTB  | 000000RG | 003 | ZF.DDM= | *****   | GX  | \$ANY =  | 000020 |    | \$NUMBR= | 000002 |    | .PSTPT=    | *****    | GX  |
| DDMSTB  | 000000RG | 002 | ZF.DIA= | *****   | GX  | \$BLANK= | 000006 |    | \$RAD50= | 000016 |    | .TPARS=    | *****    | GX  |
| FLAG    | 000000R  |     |         |         |     |          |        |    |          |        |    |            |          |     |

. ABS. 000014 000 (RW,I,GBL,ABS,OVR)  
 000260 001 (RW,I,LCL,REL,CON)  
 \$STATE 000234 002 (RW,D,LCL,REL,CON)  
 \$KIAB 000032 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000063 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

## \*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10438 Words ( 41 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:33.50

SY:TMPDDM.V2,[132,134]TMPDDM/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPDDM

SETOUT - OUTGOING TIMER

```

248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263 000214
264 000214 016700 177560
265 000220 116760 000000G 000013
266 000226 001406
267 000230 005767 000000G
268 000234 001003
269 000236 105767 000001G
270 000242 001402
271 000244 062716 000002
272 000250 000207

 .SBTTL SETOUT - OUTGOING TIMER
 +
 *** - SETOUT - SETUP OUTGOING TIMER
 THIS ACTION ROUTINE SETS UP THE OUTGOING TIMER VALUE FOR THE EXECUTOR
 INPUT:
 .PNUMB = OUTGOING TIMER
 .PNUMH = (HIGH ORDER)
 OUTPUT:
 THE OUTGOING TIMER IS STORED IN C.OUTT OF THE TEMPLATE
 -

SETOUT:
MOV TEMP,R0 ; GET START OF TEMPLATE
MOVB .PNUMB,C.OUTT(R0) ; STORE THE OUTGOING TIMER
BEQ 10$; BR IF INVALID VALUE
TST .PNUMH ; IS IT ONLY A WORD VALUE ?
BNE 10$; BR IF NO, ERROR
TSTB .PNUMB+1 ; IS IT A BYTE VALUE ?
BEQ 20$; IF EQ, YES - OKAY
ADD #2,(SP) ; ELSE, REJECT TRANSITION
10$: ADD
20$: RETURN

```

SETINA - INACTIVITY TIMER

TMPDLC - PARSE DLC PROCESS DEF1 MACRO V05.03b Saturday 29-Jun-85 05:31 <sup>F 16</sup>  
Table of contents

|     |     |                                            |
|-----|-----|--------------------------------------------|
| 6-  | 152 | SETMDLC - SETUP DLC TEMPLATE               |
| 7-  | 181 | SETNAM - SET UP PROCESS NAME               |
| 8-  | 207 | SETFLG - SET UP THE FLAGS WORD             |
| 9-  | 225 | SETPRI - SET THE PRIORITY                  |
| 10- | 251 | SETNLT - STORE NUMBER OF LINE TABLES       |
| 11- | 271 | SETNCT - STORE NUMBER OF CONTROLLER TABLES |

|                  |                  |                   |                   |                   |
|------------------|------------------|-------------------|-------------------|-------------------|
| CH.DUP= 000001   | C.LEN = 000020   | HISPD= 000005     | \$SFLG= ***** GX  | X\$2WSZ= ***** GX |
| CH.D11= 000100   | C.PRM = ***** GX | HWP 001264R       | TBLN = 000034     | X2PBLK 001466R    |
| CH.ECH= 000002   | C.STS = ***** GX | LF.ENA= ***** GX  | TBL1 001344R      | X2PBL2 001500R    |
| CH.MDT= 000020   | C.UNT 000004     | LF.MFL= ***** GX  | TBL2 001364R      | X2PBL3 001504R    |
| CH.PAR= 000100   | DDCTBL 001330R   | LF.X2P= ***** GX  | UBE 000775R       | X2PLIN 000710R    |
| CH.PRM= 000200   | DISPLY 002046R   | LINID 000226R     | UBF 001000R       | X2PMSG 000422R    |
| CH.PRT= 000007   | DPR 001164R      | LINMSG 000216R    | UBH 001003R       | X2PMS2 000474R    |
| CH.STA= 000040   | DPRVAL 001211R   | L\$CNT = ***** GX | UBJ 001006R       | X2PMS3 000545R    |
| CH.SYN= 000004   | DSPOPT 000100R   | L\$NAM = ***** GX | UBK 001011R       | \$BUFF = ***** GX |
| CH.TYP= 000010   | END 000034R      | L\$TRI = ***** GX | UBL 001014R       | \$CBDMG= ***** GX |
| CLEAR 000745R    | ERRLIS= ***** GX | L\$UNT = ***** GX | UBM 001017R       | \$CBOMG= ***** GX |
| CLEARL= 000007   | ERRX2P 000647R   | OFF 000742R       | UBN 001022R       | \$CBTA = ***** GX |
| CNTBLK 001420R   | EX\$ERR= 000002  | OFFL = 000003     | UBP 001025R       | \$CFERR= ***** GX |
| CNTMSG 000252R   | FL.MTP= ***** GX | ON 000740R        | UBR 001030R       | \$CNTNM= ***** GX |
| CPA 000761R      | FL.MUX= ***** GX | ONL = 000002      | UBS 001033R       | \$CSTA = ***** GX |
| CPB 000764R      | FL.TRB= ***** GX | OPTBLK 001446R    | UBT 001036R       | \$DVNAM= ***** GX |
| CPC 000767R      | FL.TRI= ***** GX | OPTMSG 000413R    | UNTBLK 001432R    | \$EDMSG= ***** GX |
| CPD 000772R      | FMTLIN 003426R   | OP.CNT= 000001    | UNTMSG 000337R    | \$FLAGS= ***** GX |
| CSR 001223R      | FMTLN2 003432R   | OP.OPT= 000004    | URM 001041R       | \$FNPRO= ***** GX |
| CSRVAL 001240R   | FULL 000730R     | OP.STA= 000010    | URMTBL 000036R    | \$FNSTL= ***** GX |
| CS.CNT= ***** GX | FULLN = 000004   | OP.UNT= 000002    | URMVAL 001070R    | \$FNSTA= ***** GX |
| CS.DDM= ***** GX | GETSLT 003002R   | OP.X2P= 000020    | WC.CNT= ***** GX  | \$GTBUF= ***** GX |
| CS.UNT= ***** GX | GETUNT 002260R   | PROTBL 001270R    | WC.NAM= ***** GX  | \$PRINT= ***** GX |
| CS.X2P= ***** GX | GETURM 002670R   | RATETB 000000R    | WC.UNT= ***** GX  | \$SHLIN 001510R   |
| CURSLT 000076R   | HALF 000734R     | SCRACH 001252R    | X\$2BSZ= ***** GX | \$TMLST= ***** GX |
| C.CSR 000012     | HALFLN= 000004   | SF.ENA= ***** GX  | X\$2HBT= ***** GX | \$UNTNM= ***** GX |
| C.CST 000014     | HEADR 000102R    | STABLK 001456R    | X\$2LIN= ***** GX | \$WILD = ***** GX |
| C.CWO 000006     | HEADR1 000145R   | STAMSG 000631R    | X\$2RTC= ***** GX | \$\$\$ = 000015   |
| C.CW1 000010     | HISPD 000754R    | \$SLCTT= ***** GX | X\$2RTT= ***** GX | \$\$\$1 = 000070R |
| C.DPR 000016     |                  |                   |                   |                   |

. ABS. 000020 000 (RW,I,GBL,ABS,OVR)  
003562 001 (RW,I,LCL,REL,CON)  
TEXT 000105 002 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10195 Words ( 40 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:16.39  
SY:SHOLIN.V2,[132,134]SHOLIN/CR/-SP=SY:[1,1]RSXMCML.SML/ML,[130,110]NETLIB/ML,[132,10]SHOLIN

```

253 .SBTTL PRTSNK - PRINT SINK TYPE MESSAGE
254
255 ;+
256 ; PRTSNK - PRINT SINK TYPE MESSAGE
257 ;
258 ; INPUTS:
259 ; R3 - SINK TYPE
260 ;
261 ; OUTPUTS:
262 ; THE SINK TYPE MESSAGE IS DISPLAYED
263 ;
264 ;-
265
266 000620 012702 000316' PRTSNK: MOV #MSGBLK,R2 ; POINT R2 AT ARGUMENT BLOCK
267 000624 032703 000001 BIT #FF.CON,R3 ; LOGGING CONSOLE?
268 000630 001406 BEQ 10$; BR IF NO
269 000632 012712 000007 MOV #CONL,(R2) ; STORE LENGTH OF STRING
270 000636 012762 000302' 000002 MOV #CONSL,2(R2) ; STORE ADDRESS OF STRING
271 000644 000416 BR 30$; CONTINUE
272 000646 032703 000002 10$: BIT #FF.FIL,R3 ; LOGGING FILE?
273 000652 001406 BEQ 20$; BR IF NO
274 000654 012712 000004 MOV #FILEL,(R2) ; STORE LENGTH OF STRING
275 000660 012762 000276' 000002 MOV #FILE,2(R2) ; STORE ADDRESS OF STRING
276 000666 000405 BR 30$; CONTINUE
277 000670 012712 000007 20$: MOV #MONL,(R2) ; ELSE MUST BE LOGGING MONITOR - STORE LENGTH
278 000674 012762 000267' 000002 MOV #MONITR,2(R2) ; STORE ADDRESS OF STRING
279 000702 012700 000000G 30$: MOV #$BUFF,R0 ; POINT AT OUTPUT STRING
280 000706 012701 000166' MOV #SNKMSG,R1 ; POINT AT INPUT STRING
281 000712 004767 000000G CALL $EDMSG ; FORMAT THE STRING
282 000716 PRINT #$BUFF ; PRINT THE LOGGING TYPE
283 000726 000207 RETURN

```

```

55 .SBTTL MACRO CALLS AND DEFINITIONS
56
57 : MACRO CALLS
58 :
59 .MCALL CALLR
60
61 : LOCAL MACROS
62 :
63 :
64 .MACRO PRINT TEXT
65 .IF DIF <TEXT><RO>
66 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
67 .ENDC
68 CALL $PRINT ; PRINT MESSAGE
69 .ENDM PRINT
70
71 .MACRO ERRORS$ TEXT
72 :
73 : THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
74 : MESSAGE STRING.
75 :
76 :
77 .IF DIF <TEXT><RO>
78 MOV TEXT,RO ; GET ADDRESS OF ERROR MESSAGE
79 .ENDC
80 CALL $CFERR ; PRINT ERROR MESSAGE
81 .ENDM ERRORS$
82
83 .MACRO EPRINT TEXT
84 :
85 : PRINT TEXT ON ERROR LUN
86 :
87 .IF DIF <TEXT><RO>
88 MOV TEXT,RO ; GET ADDRESS OF ASCIZ STRING TO PRINT
89 .ENDC
90 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
91 .ENDM EPRINT
92
93 :
94 : GENERATE A STRING
95 :
96 .MACRO STRING NAME,TEXT
97 NAME: .ASCII /TEXT/
98 NAME'L = . - NAME
99 .ENDM
100
101 :
102 : GENERATE TEXT AND FORMATTING TABLES
103 :
104 .MACRO BEGTXT, NAME ; START A TABLE
105 $$TXX = 0
106 $$OBIT = 2
107 .MACRO TEXT, NARG, OPT, STR ; GENERATE STRING
108 $$TXX = $$TXX + 1
109 .IF NB <OPT> 'OPT = $$OBIT
110 $$OBIT = $$OBIT+$$OBIT
111 $$$STR <STR>,NAME,\\$$TXX,\\'NARG
 .ENDM TEXT

```

```

601 .SBTTL
602 .SBTTL $SHX29 - SHOW MODULE X29-SERVER
603
604 *** - $SHX29 - SHOW MODULE X29-SERVER
605
606 INPUT:
607 $OPRND = TYPE OF REQUEST
608
609 OUTPUT:
610 THE HEADER IS PRINTED AND THE SHOW INFORMATION ROUTINE
611 IS DISPATCHED TO
612
613
614 $SHX29::
615 003464 012705 003534' MOV #X29DTB,R5 ; POINT TO DISPATCH TABLE
616 003470 012704 000000G MOV #$HDBK,R4 ; POINT TO ARGUMENT BLOCK
617 003474 012724 000012' MOV #X29L(R4)+ ; MODULE IS X29-SERVER
618 003500 012724 001706' MOV #X29L(R4)+ ;
619 003504 016703 000000G MOV $OPRND,R3 ; GET REQUEST TYPE
620 003510 042703 000000C BIC #C<X9.DST!X9.KND>,R3 ; CLEAR DONT CARE BITS
621 003514 012767 177777 174270 MOV #-1,OPTIONS ; WANT ALL OUTPUT
622 003522 012767 115447 174260 MOV #*RX29,DTYPE ; X25 DESTINATIONS ONLY
623 003530 CALLR DISPATCH ; DISPATCH TO ROUTINE
624
625 ; DISPATCH TABLE
626
627
628 003534 003566' 000000 000017 X29DTB: .NLIST BEX
629 003544 004252' 000000G 000017 .WORD X29CHA, 0, CHARL, CHAR ; CHARACTERISTICS
630 003554 004252' 000000G 000022 .WORD SHODST, X9.DST, CHARL, CHAR ; DESTINATION
631 003564 000000 .WORD SHODST, X9.KND, KDSTL, KDST ; KNOWN DESTINATIONS
632 .WORD 0
633 .LIST BEX

```



SHOMOD CREATED BY MACRO ON 29-JUN-85 AT 05:27 PAGE 2 F 5  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL | VALUE      | REFERENCES                                |
|--------|------------|-------------------------------------------|
| DSTM10 | 001505 R   | #5-230 5-231                              |
| DSTM2  | 001312 R   | #5-224 5-231                              |
| DSTM3  | 001350 R   | #5-225 5-231                              |
| DSTM4  | 001367 R   | #5-226 5-231                              |
| DSTM5  | 001405 R   | #5-227 5-231                              |
| DSTM6  | 001433 R   | #5-228 5-231                              |
| DSTM7  | 001462 R   | #5-229 5-231                              |
| DS.CH1 | = 000100   | #5-228 5-232 21-883                       |
| DS.CLO | = 000040   | #5-227 5-232 21-882                       |
| DS.CUG | = 000020   | #5-226 5-232 21-881                       |
| DS.MSK | = 000200   | #5-229 5-232 21-888                       |
| DS.NUM | = 000010   | #5-225 5-232 21-880                       |
| DS.OPT | = 000770   | #5-232 21-857                             |
| DS.VAL | = 000400   | #5-230 5-232 21-889                       |
| DTEBK  | 000000 R   | #5-202 8-373 *8-395 *8-396                |
| DTEB1  | 000000 R   | #5-202 5-202                              |
| DTEB2  | 000012 R   | #5-202 5-202                              |
| DTEB3  | 000020 R   | #5-202 5-202                              |
| DTEFT  | 001046 R   | #5-202 8-397                              |
| DTEM1  | 000724 R   | #5-199 5-202                              |
| DTEM2  | 000740 R   | #5-200 5-202                              |
| DTEM3  | 001001 R   | #5-201 5-202                              |
| DTYPE  | 000010 R   | #5-162 *12-561 *14-622 21-832 21-847      |
| DSACUG | = ***** GX | 21-881                                    |
| DSANUM | = ***** GX | 21-880                                    |
| DSASHI | = ***** GX | 21-883                                    |
| DSASLO | = ***** GX | 21-882                                    |
| DSCMSK | = ***** GX | 21-888                                    |
| DSVAL  | = ***** GX | 21-889                                    |
| DSNAM  | = ***** GX | 21-836 21-838 21-849 21-851 21-859 21-860 |
| DSOBJ  | = ***** GX | 21-867                                    |
| DSPRI  | = ***** GX | 21-861                                    |
| DSSTSK | = ***** GX | 21-871                                    |
| DSSTYP | = ***** GX | 21-832                                    |
| DSADR  | = ***** GX | 8-364 8-375 10-496                        |
| DSCT   | = ***** GX | 8-379                                     |
| DSHSH  | = ***** GX | 8-385                                     |
| DSLIN  | = ***** GX | 8-388                                     |
| DSSTA  | = ***** GX | 8-382                                     |
| ERRREQ | 000130 R   | #5-173 18-737                             |
| EXERR  | = 000002   | #5-171 5-173                              |
| FL.MTP | = ***** GX | 20-801                                    |
| FL.MUX | = ***** GX | 8-392 20-795                              |
| FL.TRB | = ***** GX | 20-803                                    |
| FMTLN2 | 004126 R   | 8-393 #20-789                             |
| FMTNRN | 000255 R   | #5-180 9-423                              |
| FMTNR1 | 000250 R   | #5-179 9-426                              |
| FMTSRN | 000265 R   | #5-181                                    |
| FMTSR1 | 000275 R   | #5-182 21-865                             |
| FMT2R5 | 000300 R   | #5-183 21-869                             |
| HDRMSG | 000213 R   | #5-178 19-758                             |
| HDRPRT | 000000 R   | #5-158 *18-722 18-732 19-755 *19-757      |

SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL  | VALUE      | REFERENCES |       |       |       |       |       |       |       |
|---------|------------|------------|-------|-------|-------|-------|-------|-------|-------|
| CLEAR   | 000266 R   | #4-104     | 4-105 | 5-155 |       |       |       |       |       |
| CLEARL  | = 000007   | #4-105     | 5-154 |       |       |       |       |       |       |
| CL.LEN  | = 000020   | #4-57      |       |       |       |       |       |       |       |
| CS.LLC  | = ***** GX | 5-162      |       |       |       |       |       |       |       |
| C.CTM   | 000016     | #4-57      |       |       |       |       |       |       |       |
| C.DEV   | 000012     | #4-57      |       |       |       |       |       |       |       |
| C.EXT   | 000015     | #4-57      |       |       |       |       |       |       |       |
| C.FLG   | 000006     | #4-57      | 5-152 |       |       |       |       |       |       |
| C.LIN   | 000011     | #4-57      |       |       |       |       |       |       |       |
| C.NAM   | 000004     | #4-57      | 5-144 |       |       |       |       |       |       |
| C.NCTD  | = 000013   | #4-57      | 5-178 |       |       |       |       |       |       |
| C.NLT   | 000014     | #4-57      | 5-167 |       |       |       |       |       |       |
| C.NLTD  | = 000012   | #4-57      | 5-176 |       |       |       |       |       |       |
| C.PRI   | 000010     | #4-57      |       |       |       |       |       |       |       |
| C.STS   | = ***** GX | 5-162      |       |       |       |       |       |       |       |
| HEADR   | 000000 R   | #4-96      | 5-124 |       |       |       |       |       |       |
| HEADR1  | 000046 R   | #4-97      | 5-127 |       |       |       |       |       |       |
| MSGBLK  | 000276 R   | #4-108     | 5-143 | 5-149 | 5-160 |       |       |       |       |
| ON      | 000264 R   | #4-102     | 4-103 | 5-151 |       |       |       |       |       |
| ONL     | = 000002   | #4-103     | 5-150 |       |       |       |       |       |       |
| O1MSG   | 000142 R   | #4-99      | 5-157 |       |       |       |       |       |       |
| O2MSG   | 000160 R   | #4-100     | 5-169 |       |       |       |       |       |       |
| O3MSG   | 000205 R   | #4-101     | 5-180 |       |       |       |       |       |       |
| PROMSG  | 000124 R   | #4-98      | 5-146 |       |       |       |       |       |       |
| WC.NAM  | = ***** GX | 5-125      |       |       |       |       |       |       |       |
| ZF.MFL  | = ***** GX | 5-152      |       |       |       |       |       |       |       |
| \$BUFF  | = ***** GX | 5-128      | 5-131 | 5-145 | 5-148 | 5-156 | 5-159 | 5-168 | 5-171 |
|         |            | 5-182      |       |       |       |       |       |       |       |
| \$EDMSG | = ***** GX | 5-130      | 5-147 | 5-158 | 5-170 | 5-181 |       |       |       |
| \$FNPRO | = ***** GX | 5-137      |       |       |       |       |       |       |       |
| \$GTBUF | = ***** GX | 5-129      |       |       |       |       |       |       |       |
| \$PRINT | = ***** GX | 5-131      | 5-148 | 5-159 | 5-171 | 5-182 |       |       |       |
| \$SHPRO | 000302 RG  | #5-123     |       |       |       |       |       |       |       |
| \$TMLST | = ***** GX | 5-132      |       |       |       |       |       |       |       |
| \$WILD  | = ***** GX | 5-125      |       |       |       |       |       |       |       |

```

402 .SBTTL GETDLL - GET DOWNLINE LOAD INFO
403 ;*
404 ** GETDLL - GET DLL INFO
405 ;
406 INPUT: R5 = TEMPLATE ADDRESS
407 ;
408 OUTPUT: DLL INFO, IF ANY, IS PRINTED
409 ;
410 ;
411 ;
412 001246 004767 000000G GETDLL: CALL $$AVAL ; save registers
413 001252 012701 000000G MOV #STMST,R1 ; point to template list
414
415 001256 011101 10$: MOV (R1),R1 ; next
416 001260 001534 BEQ 60$; if EQ, no information
417 001262 126127 000000G 000000G C.STS(R1),#CS.SER ; is it a SER$DF ?
418 001270 001372 BNE 10$; br if no
419 001272 026165 000006 000010 $ADD(R1),C.ADD(R5) ; is it for this remote ?
420 001300 001366 BNE 10$; if ne, no
421 001302 010167 177036 MOV R1,SERPT ; else save pointer
422 001306 011101 MOV (R1),R1 ; point to next guy
423 001310 126127 000000G 000000G C.STS(R1),#CS.FIL ; is this a FIL$DF ?
424 001316 001116 BNE 101$; if NE, big trouble
425 001320 010167 177022 MOV R1,FILPT ; save FIL$DF pointer
426
427 001324 012704 000340' MOV #DLLTBL~A$LEN,R4; point to control table
428 001330 012700 000000G MOV #B$UFF,R0 ; point to output buffer
429
430 001334 062704 000010 20$: ADD A$LEN,R4 ; point to next entry
431 001340 016401 000000 MOV A$STR(R4),R1 ; point to text string
432 001344 001473 BEQ 50$; if EQ, end of table
433 001346 116402 000002 MOV A$DFS(R4),R2 ; get offset
434 001352 016403 000006 MOV A$FLG(R4),R3 ; get flags
435 001356 001413 BEQ 25$; if EQ, process this one unconditionally
436 001360 132764 000100 000003 BITB #100,A$ARG(R4) ; file parameter ?
437 001366 001016 BNE 30$; if NE, yes
438 001370 016705 176750 MOV SERPT,R5 ; point to SER$DF
439 001374 001757 BEQ 20$; if EQ, no SER$DF present - keep going
440 001376 030365 000004 BIT R3,S$FLG(R5) ; is this field present in SER$DF ?
441 001402 001754 BEQ 20$; if EQ, no - keep going
442 001404 000407 BR 30$;
443
444 001406 005767 176734 25$: TST FILPT ; FIL$DF present ?
445 001412 001750 BEQ 20$; if EQ, no
446 001414 066702 176726 ADD FILPT,R2 ; point to template
447 001420 011202 MOV (R2),R2 ; point to file descriptor
448 001422 001744 BEQ 20$; if EQ, not parsed
449
450 001424 105764 000003 30$: TSTB A$ARG(R4) ; EOL before this line ?
451 001430 100011 BPL 32$; if PL, no - continue
452 001432 020027 000000G CMP R0,#B$UFF ; empty line ?
453 001436 001406 BEQ 32$; yes
454 001440 PRINT #B$UFF ; else output buffer
455 001450 012700 000000G MOV #B$UFF,R0 ; reset buffer pointer
456
457 001454 020027 000000G 32$: CMP R0,#B$UFF ; first parameter on text line ?
458 001460 001405 BEQ 35$; if EQ, yes

```

.TITLE SHOSYS - CFE SHOW SYSTEM PARAMETERS  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

# MODULE DESCRIPTION:

CFE - SHOW SYSTEM DEFAULTS

## DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

.MACRO PRINT TEXT  
MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT  
CALL \$PRINT ; PRINT MESSAGE

```

135 .SBTTL TPARS STATE TABLES
136 ;
137 ; INITIALIZE TABLES
138 ;
139 ;
140 000034 ISTAT$ BUFSTB,BUFKTB
141 ;
142 ; PROCESS 'BUF$DF'
143 000034 STATES$
144 000034 TRANS$ $NUMBR,,SETCBN
145 ;
146 000034 STATES$
147 000034 TRANS$ '<','>'
148 ;
149 000034 STATES$
150 000034 TRANS$ $NUMBR,,SETCBS
151 ;
152 000034 STATES$
153 000034 TRANS$ '<','>'
154 ;
155 000034 STATES$
156 000034 TRANS$ $NUMBR,,SETRBN
157 ;
158 000034 STATES$
159 000034 TRANS$ '<','>'
160 ;
161 000034 STATES$
162 000034 TRANS$ $NUMBR,,SETRBS
163 ;
164 000034 STATES$
165 000034 TRANS$ '<','>'
166 ;
167 000034 STATES$
168 000034 TRANS$ $NUMBR,,SETSBN
169 ;
170 000034 STATES$
171 000034 TRANS$ '<','>'
172 ;
173 000034 STATES$
174 000034 TRANS$ $NUMBR,,SETSBS
175 ;
176 000034 STATES$
177 000034 TRANS$ '<','>'
178 ;
179 000034 STATES$
180 000034 TRANS$ $NUMBR,,SETTHR
181 ;
182 000034 STATES$
183 000034 TRANS$ $EOS,$EXIT
184 000034 TRANS$ $LAMBDA,$EXIT
185 ;
186 ; FINAL STATE
187 ;
188 000034 STATES$

```

TMPBUF      CREATED BY    MACRO    ON 29-JUN-85 AT 05:28      PAGE 3      F 10

MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$ | #6-140 | #6-184 |        |        |        |        |        |        |        |        |
| EPRINT  | #4-77  |        |        |        |        |        |        |        |        |        |
| ERROR\$ | #4-66  |        |        |        |        |        |        |        |        |        |
| ISTAT\$ | #4-90  | 6-140  |        |        |        |        |        |        |        |        |
| MTRAN\$ | #6-140 |        |        |        |        |        |        |        |        |        |
| PRINT   | #4-61  |        |        |        |        |        |        |        |        |        |
| STAT\$  | #4-90  | 6-143  | #6-146 | #6-149 | #6-152 | #6-155 | #6-158 | #6-161 | #6-164 | #6-167 |
|         | #6-170 | #6-173 | #6-176 | #6-179 | #6-182 | #6-188 |        |        |        |        |
|         | #4-90  | #6-144 | #6-147 | #6-150 | #6-153 | #6-156 | #6-159 | #6-162 | #6-165 | #6-168 |
| TRAN\$  | #6-171 | #6-174 | #6-177 | #6-180 | #6-183 | #6-184 |        |        |        |        |

\*\*FILE\*\*ID\*\*TMPONT

```

TTTTTTTTTT MM MM PPPPPPP CCCCCCCC NN NN TTTTTTTTTT
TTTTTTTTTT MM MM PPPPPPP CCCCCCCC NN NN TTTTTTTTTT
TT MMMM MMMM PP CC NN NN TT
TT MMMM MMMM PP CC NN NN TT
TT MM MM MM PP CC NNNN NN TT
TT MM MM MM PP CC NNNN NN TT
TT MM MM PPPPPPP CC NN NN NN TT
TT MM MM PPPPPPP CC NN NN NN TT
TT MM MM PP CC NN NNNN TT
TT MM MM PP CC NN NNNN TT
TT MM MM PP CC NN NN TT
TT MM MM PP CC NN NN TT
TT MM MM PP CCCCCCCC NN NN TT
TT MM MM PP CCCCCCCC NN NN TT
TT MM MM PP

```

```

....
....
....
....

```

```

LL SSSSSSS TTTTTTTTTT
LL SSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLLL SSSSSSS TT
LLLLLLLLLLL SSSSSSS TT

```

IMPCNT      CREATED BY    MACRO    ON 29-JUN-85 AT 05:29      PAGE 2      F 12

MACRO CROSS REFERENCE

CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$ | #5-95  | #5-123 |        |        |        |        |        |        |        |        |
| ISTAT\$ | #4-68  | 5-95   |        |        |        |        |        |        |        |        |
| MTRAN\$ | #5-95  |        |        |        |        |        |        |        |        |        |
| PRINT   | #4-59  |        |        |        |        |        |        |        |        |        |
| STAT\$  | #4-68  | 5-97   | #5-100 | #5-103 | #5-106 | #5-109 | #5-112 | #5-115 | #5-118 | #5-121 |
|         | #5-125 | #5-128 | #5-130 | #5-133 | #5-136 |        |        |        |        |        |
| TRAN\$  | #4-68  | #5-98  | #5-101 | #5-104 | #5-107 | #5-110 | #5-113 | #5-116 | #5-119 | #5-122 |
|         | #5-123 | #5-126 | #5-129 | #5-131 | #5-134 |        |        |        |        |        |



[illegible]

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
 SS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

TMPDDM      CREATED BY MACRO ON 29-JUN-85 AT 05:30      PAGE 1      F 14  
 SYMBOL CROSS REFERENCE      CREF      04.00

| SYMBOL    | VALUE       | REFERENCES                                    |
|-----------|-------------|-----------------------------------------------|
| CS.DDM    | = ***** GX  | 6-176                                         |
| C.FLG     | = 000006    | #4-84 *8-228                                  |
| C.LEN     | = 000014    | #4-88 6-173                                   |
| C.NAM     | = 000004    | #4-83 *7-207                                  |
| C.NCT     | = 000013    | #4-87 *11-291                                 |
| C.NLT     | = 000012    | #4-86 *6-177                                  |
| C.PRI     | = 000010    | #4-85 *9-249                                  |
| C.STS     | = ***** GX  | *10-271                                       |
| DDMKTB    | = 000000 RG | *6-176                                        |
| DDMSTB    | = 000000 RG | #5-95 6-179                                   |
| FLAG      | = 000000 R  | #5-95 6-180                                   |
| SETFLG    | = 000124 R  | #4-74 8-228                                   |
| SETNAM    | = 000064 R  | #8-226                                        |
| SETNCT    | = 000230 R  | #7-202                                        |
| SETNLT    | = 000200 R  | #11-289                                       |
| SETPRI    | = 000140 R  | #10-269                                       |
| TEMP      | = 000002 R  | #9-247                                        |
| \$ALPHA   | = 000022    | #4-75 *6-175 7-206 8-227 9-248 10-270 11-290  |
| \$ANY     | = 000020    | #5-95                                         |
| \$BLANK   | = 000006    | #5-95                                         |
| \$CAT5    | = ***** GX  | #5-95                                         |
| \$DIGIT   | = 000024    | 7-205                                         |
| \$DNUMB   | = 000014    | #5-95                                         |
| \$EOS     | = 000012    | #5-95                                         |
| \$ERROR   | = ***** GX  | *6-183                                        |
| \$EXIT    | = 000000    | #5-95                                         |
| \$FAIL    | = 177777    | #5-95                                         |
| \$GPRM    | = *****     | 5-95                                          |
| \$LAMDA   | = 000000    | #5-95                                         |
| \$MXNCT   | = ***** GX  | 11-292                                        |
| \$MXNLT   | = ***** GX  | 10-272                                        |
| \$NUMBR   | = 000002    | #5-95                                         |
| \$RAD50   | = 000016    | #5-95                                         |
| \$RONLY   | = *****     | 5-95 5-95                                     |
| \$STRNG   | = 000004    | #5-95                                         |
| \$SUE:P   | = 000010    | #5-95                                         |
| \$TALOC   | = ***** GX  | 6-174                                         |
| \$TMDDM   | = 000004 RG | #6-171                                        |
| \$\$\$FLG | = 177777    | #5-95                                         |
| \$\$\$KEY | = 177777    | #5-95                                         |
| .PNUMB    | = ***** GX  | 9-249 9-250 9-252 10-271 10-272 11-291 11-292 |
| .PSTCN    | = ***** GX  | 7-208                                         |
| .PSTPT    | = ***** GX  | 7-203                                         |
| .TPARS    | = ***** GX  | 6-181                                         |

TMPDEC - CFE PARSE DEC\$DF DEFIN MACRO V05.03b Saturday 29-Jun-85 05:30 Page 11  
 SETINA - INACTIVITY TIMER

```

274 .SBTTL SETINA - INACTIVITY TIMER
275
276 ;+
277 ;*** - SETINA - SETUP INACTIVITY TIMER
278 ;THIS ACTION ROUTINE SETS UP THE INACTIVITY TIMER VALUE FOR THE EXECUTOR
279
280 ;INPUT:
281 ; .PNUMB = INACTIVITY TIMER
282 ; .PNUMH = (HIGH ORDER)
283
284 ;OUTPUT:
285 ; THE INACTIVITY TIMER IS STORED IN C.INAC OF THE TEMPLATE
286
287 ;-
288
289 SETINA:
290 000252 016700 177522 MOV TEMP,RO ; GET START OF TEMPLATE
291 000256 116760 000000G 000014 MOVB .PNUMB,C.INAC(RO) ; STORE THE INACTIVITY TIMER
292 000264 001406 BEQ 10$; BR IF INVALID VALUE
293 000266 005767 000000G TST .PNUMH ; IS IT ONLY A WORD VALUE ?
294 000272 001003 BNE 10$; BR IF NO - ERROR
295 000274 105767 000001G TSTB .PNUMB+1 ; IS IT A BYTE VALUE ?
296 000300 001402 BEQ 20$; IF EQ, YES - OKAY
297 000302 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT TRANSITION
298 000306 000207 20$: RETURN

```

TMPDEC - CFE PARSE DEC\$DF DEFIN MACRO V05.03b Saturday 29-Jun-85 05:30 Page 12  
 SETDEF - DELAY FACTOR

.TITLE TMPDLC - PARSE DLC PROCESS DEFINITION IN CERAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE DLC DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0

3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1

4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0

5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/Rsx V1.0

SHOLIN      CREATED BY    MACRO    ON 29-JUN-85 AT 05:26      PAGE 1      G 1  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE      | REFERENCES                                                    |
|---------|------------|---------------------------------------------------------------|
| CH.DUP  | = 000001   | 7-447                                                         |
| CH.MDT  | = 000020   | 7-486                                                         |
| CH.TYP  | = 000010   | 7-483                                                         |
| CLEAR   | = 000745 R | #4-214 4-215 9-627                                            |
| CLEARL  | = 000007   | #4-215 9-626                                                  |
| CNTBLK  | 001420 R   | 4-269 #4-277 *5-335 *5-336 *5-337 *8-536 *8-537 *8-551 *8-552 |
| CNTMSG  | 000252 R   | #4-193 4-260                                                  |
| CPA     | 000761 R   | 4-146 #4-220                                                  |
| CPB     | 000764 R   | 4-147 #4-221                                                  |
| CPC     | 000767 R   | 4-148 #4-222                                                  |
| CPD     | 000772 R   | 4-149 #4-223                                                  |
| CSR     | 001223 R   | #4-240 *7-441 *7-505 7-512                                    |
| CSRVAL  | 001240 R   | #4-241 7-513                                                  |
| CS.CNT  | = ***** GX | 5-325 5-349                                                   |
| CS.DDM  | = ***** GX | 5-314                                                         |
| CS.UNT  | = ***** GX | 5-345                                                         |
| CS.X2P  | = ***** GX | 9-607                                                         |
| CURSLT  | 000076 R   | #4-163 *5-319 9-589 *9-595                                    |
| C.CSR   | 000012     | #4-175 7-510                                                  |
| C.CST   | 000014     | #4-176                                                        |
| C.CWO   | 000006     | #4-173 7-447 7-456                                            |
| C.CW1   | 000010     | #4-174 7-475 7-483 7-486                                      |
| C.DPR   | 000016     | #4-177 7-496                                                  |
| C.LEN   | = 000020   | #4-178                                                        |
| C.PRM   | = ***** GX | 5-316 5-318 5-332 5-334 5-335 5-336 5-337 5-356 5-358         |
|         |            | 8-538 9-643 9-645                                             |
| C.STS   | = ***** GX | 5-314 5-325 5-349 9-607                                       |
| C.UNT   | 000004     | #4-172                                                        |
| DDCTBL  | 001330 R   | #4-255 7-491 7-492                                            |
| DISPLY  | 002046 R   | 5-360 #6-379                                                  |
| DPR     | 001164 R   | #4-238 7-498                                                  |
| DPRVAL  | 001211 R   | #4-239 7-499                                                  |
| DSPOPT  | 000100 R   | #4-164 6-391 *7-435 *7-506 *7-518 *9-615                      |
| END     | 000034 R   | #4-138 4-140                                                  |
| ERRLIS  | = ***** GX | 6-417                                                         |
| ERRX2P  | 000647 R   | #4-201 9-625                                                  |
| EX\$ERR | = 000002   | #4-183 4-201                                                  |
| FL.MTP  | = ***** GX | 6-396 9-652 10-689                                            |
| FL.MUX  | = ***** GX | 5-352 10-683                                                  |
| FL.TPB  | = ***** GX | 6-402 6-412 10-691                                            |
| FL.TRI  | = ***** GX | 5-307 6-400 9-638 9-654                                       |
| FMILIN  | 003426 R   | 6-382 6-403 #10-675                                           |
| FMTLN2  | 003432 R   | 9-624 #10-676                                                 |
| FULL    | 000730 R   | #4-206 4-207 7-446                                            |
| FULLN   | = 000004   | #4-207 7-445                                                  |
| GETSLT  | 003002 R   | 6-380 #9-586                                                  |
| GETUNT  | 002260 R   | 5-359 #7-435                                                  |
| GETURM  | 002670 R   | 5-338 #8-535                                                  |
| HALF    | 000734 R   | #4-208 4-209 7-450                                            |
| HALFLN  | = 000004   | #4-209 7-449                                                  |
| HEADR   | 000102 R   | #4-189 5-298                                                  |
| HEADR1  | 000145 R   | #4-190 5-301                                                  |

```

285 .SBTTL PRTSTA - PRINT LOGGING STATE
286
287 :+
288 :
289 : PRTSTA - PRINT LOGGING STATE
290 :
291 : INPUTS:
292 : STATE - LOGGING STATE WORD
293 : R3 - LOGGING SINK TYPE
294 :
295 : OUTPUTS:
296 : THE LOGGING STATE MESSAGE IS DISPLAYED
297 :
298 :-
299 000730 012702 000316' PRTSTA: MOV #MSGBLK,R2 ; POINT R2 AT ARGUMENT BEVTK
300 000734 012712 000003' MOV #OFFL(R2) ; ASSUME SINK TYPE IS OFF
301 000740 012762 000264' 000002 MOV #OFF.2(R2) ;
302 000746 130367 177343 BITB R3,STATE ; IS THIS SINK TYPE ON?
303 000752 001405 BEQ 10$; IF EQ, NO : OKAY
304 000754 012712 000002' MOV #ONL(R2) ; ELSE, CHANGE MESSAGE
305 000760 012762 000262' 000002 MOV #ON.2(R2) ;
306 000766 012700 000000G 10$: MOV #$BUFF,R0 ; POINT R0 AT OUTPUT BUFFER
307 000772 012701 000230' MOV #STAMSG,R1 ; POINT R1 AT INPUT STRING
308 000776 004767 000000G CALL $EDMSG ; EDIT THE STRING
309 001002 PRINT #$BUFF ; PRINT THE LINE
310 001012 000207 RETURN

```

```

112 .MACRO ENDTXT ; BUILD ARG BLOCK AND FORMATTING TABLE
113 .PSECT ARGBLK,D,OVR
114 .IIF 'NDF' $$BASE $$BASE=.
115 .=$$BASE
116 'NAME'BK:
117 $$TTX1 = 1
118 .REPT $$TTX0
119 $$$BLK NAME,\$$TTX1
120 $$TTX1 = $$TTX1 + 1
121 .ENDR
122 .PSECT
123 .EVEN
124 'NAME'FT:
125 $$TTX1 = 1
126 .REPT $$TTX0
127 $$$FMT NAME,\$$TTX1
128 $$TTX1 = $$TTX1 + 1
129 .ENDR
130 .WORD 0
131 .ENDM ENDTXT
132 .ENDM BEGTXT
133
134 .MACRO $$$STR TEXT,NAM,NUM,NUM2
135 'NAM'M'NUM: .ASCIIZ /TEXT/
136 $$TT'NUM = 'NUM2
137 .ENDM $$$STR
138
139 .MACRO $$$BLK NAM,NUM
140 'NAM'B'NUM: .BLKW $$TT'NUM
141 .ENDM $$$BLK
142
143 .MACRO $$$FMT NAM,NUM
144 .WORD 'NAM'M'NUM,'NAM'B'NUM
145 .ENDM $$$FMT
146
147 .MACRO OPTION OFFS,WC,CSBIT,DSBIT
148 JSR R4,MOVOP
149 .WORD OFFS,WC,DSBIT,CSBIT
150 .ENDM
151

```

```

635 .SBTTL X29CHA - SHOW X29 SERVER CHARACTERISTICS
636
637 ;+
638 ;** X29CHA - SHOW X29 SERVER CHARACTERISTICS
639 ;
640 ; INPUT X29$DF TEMPLATE (IF ANY).
641 ;
642 ; OUTPUT THE X29 SERVER CHARACTERISTICS ARE DISPLAYED.
643 ;
644 X29CHA:
645 003566 012700 000000G MOV #$TMLST,R0 ; POINT TO TEMPLATES
646 003572 011000 10$: MOV (R0),R0 ; NEXT TEMPLATE
647 003574 001416 BEQ 20$; IF EQ, NO INFORMATION
648 003576 122760 000000G 000000G CMPB #CS.X29,C.STS(R0) ; CHECK FOR X29 INFORMATION
649 003604 001372 BNE 10$; IF NE, CONTINUE
650 003606 012702 000000' MOV #XSVCBK,R2 ; POINT TO ARGUMENT LIST
651 003612 016022 000000G MOV X$9MXC(R0),(R2)+ ; MAXIMUM CIRCUITS
652 003616 016022 000000G MOV X$9CT(R0),(R2)+ ; COUNTER TIMER
653 003622 012703 001260' MOV #XSVCFT,R5 ; POINT TO FORMATTING TABLE
654 003626 004767 000200 CALL PRINT ; SHOW INFORMATION
655 003632 000207 20$: RETURN

```



SHOMOD CREATED BY MACRO ON 29-JUN-85 AT 05:27 PAGE 3 G 5  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL  | VALUE      | REFERENCES                                                   |
|---------|------------|--------------------------------------------------------------|
| KCUG    | 001773 R   | #5-249 5-249 6-294                                           |
| KCUGL   | = 000014   | #5-249 6-294                                                 |
| KDST    | 001737 R   | #5-247 5-247 11-536 12-569 14-630                            |
| KDSTL   | = 000022   | #5-247 11-536 12-569                                         |
| KDTE    | 001761 R   | #5-248 5-248 6-292                                           |
| KDTEL   | = 000012   | #5-248 6-292                                                 |
| KNOWN   | 000006 R   | #5-161                                                       |
| LINBUF  | 000034 R   | #5-165 8-386 8-396                                           |
| LINHDR  | 000002 R   | #5-159 *9-416 9-431 *9-451                                   |
| L\$CNT  | = ***** GX | 20-792                                                       |
| L\$LCMT | = ***** GX | 13-595                                                       |
| L\$LNAM | = ***** GX | 13-593                                                       |
| L\$NAM  | = ***** GX | 20-789                                                       |
| L\$TRI  | = ***** GX | 20-806                                                       |
| L\$UNT  | = ***** GX | 20-798                                                       |
| MOVOP   | 004632 R   | 21-880 21-881 21-882 21-883 21-888 21-889 #22-917            |
| NOINF   | 000171 R   | #5-177 18-734                                                |
| OBUF    | 000004 R   | #5-160 *9-417 9-428 *9-452 9-455                             |
| OPTION  | 000012 R   | #5-163 *6-284 *11-528 *12-560 *14-621 19-765 *21-857 *22-927 |
| PRCHBK  | 000000 R   | #5-194 7-314                                                 |
| PRCHB1  | 000000 R   | #5-194 5-194                                                 |
| PRCHB2  | 000004 R   | #5-194 5-194                                                 |
| PRCHB3  | 000010 R   | #5-194 5-194                                                 |
| PRCHB4  | 000014 R   | #5-194 5-194                                                 |
| PRCHB5  | 000020 R   | #5-194 5-194                                                 |
| PRCHB6  | 000024 R   | #5-194 5-194                                                 |
| PRCHFT  | 000672 R   | #5-194 7-338                                                 |
| PRCHM1  | 000304 R   | #5-188 5-194                                                 |
| PRCHM2  | 000355 R   | #5-189 5-194                                                 |
| PRCHM3  | 000430 R   | #5-190 5-194                                                 |
| PRCHM4  | 000500 R   | #5-191 5-194                                                 |
| PRCHM5  | 000554 R   | #5-192 5-194                                                 |
| PRCHM6  | 000624 R   | #5-193 5-194                                                 |
| PRINT   | 004032 R   | 7-339 8-398 10-507 13-597 15-654 17-700 #19-754 21-894       |
| PRTCN   | 002560 R   | 8-399 #9-414 7-335                                           |
| P\$SNAM | = ***** GX | 7-334                                                        |
| P\$SPOR | = ***** GX | 13-586                                                       |
| R\$DNAM | = ***** GX | 17-689                                                       |
| SCRBLK  | 000014 R   | #5-164                                                       |
| SHODST  | 004252 R   | 12-568 12-569 14-629 14-630 #21-822                          |
| SHOXAD  | 003636 R   | 11-535 11-536 #17-681                                        |
| SPCDST  | 002024 R   | #5-259 21-843                                                |
| TYPB    | 002007 R   | #5-250 5-250 10-505                                          |
| TYPBL   | = 000013   | #5-250 10-504 11-525                                         |
| XAC     | 001674 R   | #5-244 5-244                                                 |
| XACCHA  | 003634 R   | 11-534 #16-669                                               |
| XACDTB  | 003244 R   | 11-522 #11-534                                               |
| XACL    | = 000012   | #5-244 11-524                                                |
| XADBK   | 000000 R   | #5-238 17-693                                                |
| XADB1   | 000000 R   | #5-238 5-238                                                 |
| XADFT   | 001640 R   | #5-238 17-699                                                |
| XADM1   | 001574 R   | #5-237 5-238                                                 |

SHOPRO      CREATED BY MACRO   ON 29-JUN-85 AT 05:27      PAGE 2      G 6  
MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|        |       |       |       |       |       |       |
|--------|-------|-------|-------|-------|-------|-------|
| EPRINT | #4-80 |       |       |       |       |       |
| ERRORS | #4-69 |       |       |       |       |       |
| LLCTPS | #4-55 | 4-57  |       |       |       |       |
| PRINT  | #4-62 | 5-131 | 5-148 | 5-159 | 5-171 | 5-182 |

|     |        |        |         |        |        |                     |                         |
|-----|--------|--------|---------|--------|--------|---------------------|-------------------------|
| 459 | 001462 | 112720 | 000054  |        | MOVB   | #<'>,(R0)+          | : move over             |
| 460 | 001466 | 112720 | 000040  |        | MOVB   | #<'>,(R0)+          | : ...                   |
| 461 | 001472 | 000405 |         |        | BR     | 40\$                | : continue              |
| 462 |        |        |         |        |        |                     |                         |
| 463 | 001474 | 112720 | 000040  | 35\$:  | MOVB   | #<'>,(R0)+          | : tab over ...          |
| 464 | 001500 | 120027 | 000022G |        | CMPB   | R0,#\$BUFF+NAMCOL+3 | : ...                   |
| 465 | 001504 | 003773 |         |        | BLE    | 35\$                | : ...                   |
| 466 |        |        |         |        |        |                     |                         |
| 467 | 001506 | 112120 |         | 40\$:  | MOVB   | (R1)+,(R0)+         | : move text string      |
| 468 | 001510 | 001376 |         |        | BNE    | 40\$                | : if NE, more           |
| 469 | 001512 | 005300 |         |        | DEC    | R0                  | : back over NUL         |
| 470 | 001514 | 116403 | 000003  |        | MOVB   | A\$ARG(R4),R3       | : pick up argument      |
| 471 | 001520 | 042703 | 177700  |        | BIC    | #^C77,R3            | : remove EOL flag bit   |
| 472 | 001524 | 004774 | 000004  |        | CALL   | @A\$RTN(R4)         | : format parameter      |
| 473 | 001530 | 105010 |         |        | CLRB   | (R0)                | : zero terminate buffer |
| 474 | 001532 | 000700 |         |        | BR     | 20\$                | : continue              |
| 475 |        |        |         |        |        |                     |                         |
| 476 | 001534 | 020027 | 000000G | 50\$:  | CMP    | R0,#\$BUFF          | : empty buffer ?        |
| 477 | 001540 | 001404 |         |        | BEQ    | 60\$                | : if EQ, yes            |
| 478 | 001542 |        |         |        | PRINT  | #\$BUFF             | : flush buffer          |
| 479 | 001552 | 000207 |         | 60\$:  | RETURN |                     |                         |
| 480 |        |        |         |        |        |                     |                         |
| 481 | 001554 | 000167 | 000000G | 101\$: | JMP    | .CONER              | : big trouble somewhere |
| 482 |        |        |         |        |        |                     |                         |

```

58 .ENDM PRINT
59
60 .MACRO ERRORS$ TEXT
61 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
62 ; MESSAGE STRING.
63 ;
64 .IF DIF <TEXT><R0>
65 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
66 .ENDC
67 CALL $CFERR ; PRINT ERROR MESSAGE
68 .ENDM ERRORS$
69
70 .MACRO EPRINT TEXT
71 ; PRINT TEXT ON ERROR LUN
72 ;
73 .IF DIF <TEXT><R0>
74 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
75 .ENDC
76 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
77 .ENDM EPRINT
78
79 ;****
80 ; LOCAL DATA
81 ;****
82
83 ;
84 ; DEFINE PARTITION TEMPLATE
85 ;
86 .ASECT
87 .=0
88 .BLKW 1 ; C.LNK
89 .BLKW 1 ; C.STS
90 .BLKW 2 ; POOL PARTITION NAME (RAD50)
91 .BLKW 2 ; DYNAMIC PARTITION TO CREATE POOL FROM
92 .BLKW 1 ; POOL EXTENSION SIZE
93 .BLKW 1 ; NUMBER OF BLOCKS FOR SINGLE WORD POOL
94 .BLKW 1
95 .LEN=
96 .000020
97
98 ; DEFINE BUFFER TEMPLATE
99 ;
100 .=0
101 .BLKW 1 ; C.LNK
102 .BLKW 1 ; C.STS
103 .BLKW 1 ; CCB NUMBER
104 .BLKW 1 ; CCB SIZE (FIXED)
105 .BLKW 1 ; RDB NUMBER
106 .BLKW 1 ; RDB SIZE
107 .BLKW 1 ; SDB NUMBER
108 .BLKW 1 ; SDB SIZE
109 .BLKW 1 ; RDB THRESHOLD VALUE
110 .PSECT
111
112 .NLIST BEX
113 .ENABL LC
114 EX$ERR = 2

```

```

190 .SBTTL $TMBUF - BUILD BUFFER TEMPLATE
191
192 *** - $TMBUF - BUILD BUFFER TEMPLATE
193
194 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
195
196 INPUT:
197 R3-R5 - TPARS REGISTERS
198
199 OUTPUT:
200 C-BIT = SUCCESS/FAILURE
201 THE BUFFER TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
202
203
204
205 $TMBUF::
206 MOV R5, -(SP) ; SAVE R5
207 MOV #C.LEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
208 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
209 MOV R0, TEMP ; SAVE TEMPLATE ADDRESS
210 MOV #CS.BUF, C.STS(R0) ; INDICATE THAT THIS IS A BUF$DF TEMPLATE
211 CLR R1 ; IGNORE BLANKS
212 MOV #BUF.KTB, R2 ; KEYWORD TABLE
213 MOV #BUF.STB, R5 ; STATE TABLE
214 CALL TPARS ; PARSE THE REST OF THE LINE
215 BCC 20$; NORMAL RETURN IF NO ERROR
216 INC $ERROR ; INDICATE SYNTAX ERROR
217
218 ; INITIALIZE THE BLOCK-AREA ALLOCATION
219
220 20$: MOV TEMP, R5 ; POINT TO TEMPLATE
221 CLR $BLKMN ; NO ALLOCATION YET
222 CLR $BLKMN+2
223 MOV C.RBNM(R5), R0 ; GET NUMBER OF RDBS
224 MOV C.RBSZ(R5), R1 ; RDB SIZE
225 CALL UPDALC ; UPDATE ALLOCATION
226 MOV C.SBNM(R5), R0 ; GET NUMBER OF SDBS
227 MOV C.SBSZ(R5), R1 ; SDB SIZE
228 CALL UPDALC ; UPDATE ALLOCATION
229 MOV $BLKMN, $BLKXT ; MINIMUM IS EXTENSION
230 MOV $BLKMN+2, $BLKXT+2
231 MOV (SP)+, R5 ; RESTORE R5
232 RETURN

```

\*\*FILE\*\*ID\*\*TMPCHN

```

TTTTTTTTTT MM MM PPPPPPP CCCCCCCC HH HH NN NN
TTTTTTTTTT MM MM PPPPPPP CCCCCCCC HH HH NN NN
TT MMMM MMMM PP PP CC HH HH NN NN
TT MMMM MMMM PP PP CC HH HH NN NN
TT MM MM MM PP PP CC HH HH NNNN NN
TT MM MM PP PP CC HH HH NNNN NN
TT MM MM PPPPPPP CC HHHHHHHHHH NN NN NN
TT MM MM PPPPPPP CC HHHHHHHHHH NN NN NN
TT MM MM PP CC HH HH NN NNNN
TT MM MM PP CC HH HH NN NNNN
TT MM MM PP CC HH HH NN NN
TT MM MM PP CCCCCCCC HH HH NN NN
TT MM MM PP CCCCCCCC HH HH NN NN

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

IMPCNT - PARSE CONTROLLER DEFIN MACRO V05.03b Saturday 29-Jun-85 05:29 <sup>H 11</sup>  
Table of contents

|     |     |                                     |
|-----|-----|-------------------------------------|
| 6-  | 138 | \$TMCNT - SETUP CONTROLLER TEMPLATE |
| 7-  | 167 | SETCNT - SET UP CONTROLLER NUMBER   |
| 8-  | 192 | SETVEC - SETUP THE VECTOR ADDRESS   |
| 9-  | 218 | SETCSR - SET THE CSR ADDRESS        |
| 10- | 242 | SETPRI - SET THE PRIORITY           |
| 11- | 269 | SETURM - SET THE UNIBUS RUN MASK    |

\*\*FILE\*\*ID\*\*TMPUG

```

TTTTTTTTTT MM MM PPPPPPP CCCCCCCC UU UU GGGGGGGG
TTTTTTTTTT MM MM PPPPPPP CCCCCCCC UU UU GGGGGGGG
TT MM MM PP PP CC UU UU GG
TT MMMM MMMM PP PP CC UU UU GG
TT MM MM PP PP CC UU UU GG
TT MM MM PP PP CC UU UU GG
TT MM MM PPPPPPP CC UU UU GG
TT MM MM PPPPPPP CC UU UU GG
TT MM MM PP CC UU UU GG GGGGGG
TT MM MM PP CC UU UU GG GGGGGG
TT MM MM PP CC UU UU GG GG
TT MM MM PP CC UU UU GG GG
TT MM MM PP CCCCCCCC UUUUUUUUU GGGGGG
TT MM MM PP CCCCCCCC UUUUUUUUU GGGGGG

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```



TMPDDM - PARSE DEVICE PROCESS D MACRO V05.03b Saturday 29-Jun-85 05:30 <sup>H 13</sup>  
Table of contents

|     |     |                                            |
|-----|-----|--------------------------------------------|
| 6-  | 158 | \$TMDDM - SETUP DDM TEMPLATE               |
| 7-  | 187 | SETNAM - SET UP PROCESS NAME               |
| 8-  | 213 | SETFLG - SETUP THE FLAGS WORD              |
| 9-  | 231 | SETPRI - SET THE PRIORITY                  |
| 10- | 257 | SETNLT - STORE NUMBER OF LINE TABLES       |
| 11- | 277 | SETNCT - STORE NUMBER OF CONTROLLER TABLES |

TMPDDM      CREATED BY    MACRO    ON 29-JUN-85 AT 05:30      PAGE 2      G 14  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-95      | #5-111 | #5-118 | #5-119 | #5-120 | #5-121 | #5-122 | #5-123 | #5-124 | #5-125 |
|            | #5-126     | #5-127 | #5-137 | #5-141 | #5-145 |        |        |        |        |        |
| ISTAT\$    | #4-68      | 5-95   |        |        |        |        |        |        |        |        |
| MTRAN\$    | #5-95      |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-59      |        |        |        |        |        |        |        |        |        |
| STAT\$     | #4-68      | 5-97   | #5-100 | #5-103 | #5-106 | #5-109 | #5-113 | #5-116 | #5-129 | #5-132 |
|            | #5-135     | #5-139 | #5-143 | #5-147 | #5-150 | #5-153 | #5-156 |        |        |        |
| TRAN\$     | #4-68      | #5-98  | #5-101 | #5-104 | #5-107 | #5-110 | #5-111 | #5-114 | #5-117 | #5-118 |
|            | #5-119     | #5-120 | #5-121 | #5-122 | #5-123 | #5-124 | #5-125 | #5-126 | #5-127 | #5-130 |
|            | #5-133     | #5-136 | #5-137 | #5-140 | #5-141 | #5-144 | #5-145 | #5-148 | #5-151 | #5-154 |

```

300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
000310
000310 016700 177464
000314 116760 000000G 000016
000322 005767 000000G
000326 001003
000330 105767 000001G
000334 001402
000336 062716 000002
000342 000207

.SBTTL SETDEF - DELAY FACTOR
*** - SETDEF - SETUP DELAY FACTOR
THIS ACTION ROUTINE SETS UP THE DELAY FACTOR FOR THE EXECUTOR
INPUT:
.PNUMB = DELAY FACTOR
.PNUMH = (HIGH ORDER)
OUTPUT:
THE DELAY FACTOR IS STORED IN C.DELF OF THE TEMPLATE

SETDEF:
MOV TEMP,RO ; GET START OF TEMPLATE
MOVB .PNUMB,C.DELF(RO) ; STORE DELAY FACTOR
TST .PNUMH ; IS IT ONLY A WORD VALUE ?
BNE 10$; IF NE, NO - REJECT
TSTB .PNUMB+1 ; IS IT A BYTE VALUE ?
BEQ 20$; IF EQ, YES - OKAY
10$: ADD #2,(SP) ; ELSE, REJECT TRANSITION
20$: RETURN

```

```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 ;****
65 ; MACRO CALLS
66 ;****
67
68 .MCALL ISTAT$,STATE$,TRANS$
69
70 ;****
71 ; LOCAL DATA
72 ;****
73
74 000000 FLAG: .BLKW 1 ; FLAGS WORD
75 000002 TEMP: .BLKW 1 ; ADDRESS OF DLC TEMPLATE
76
77 ; DEFINE DLC TEMPLATE
78 ;
79 000000 .ASECT
80 000000 .=0
81 000000 .BLKW 1 ; C.LNK
82 000002 .BLKW 1 ; C.STS
83 000004 C.NAM: .BLKW 1 ; DLC PROCESS NAME
84 000006 C.FLG: .BLKW 1 ; DLC FLAGS WORD
85 000010 C.PRI: .BLKW 1 ; DLC PROCESS PRIORITY
86 000012 C.NLT: .BLKB 1 ; NUMBER OF LINE TABLES IN PROCESS SPACE
87 000013 C.NCT: .BLKB 1 ; NUMBER OF CONTROLLER TABLES IN PROCESS SPACE
88 000014 C.LEN=.
89 000004 .PSECT
90

```

SHOLIN CREATED BY MACRO ON 29-JUN-85 AT 05:26 PAGE 2 H 1  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL   | VALUE      | REFERENCES                                                    |
|----------|------------|---------------------------------------------------------------|
| HISPD    | 000754 R   | #4-216 4-217 7-455                                            |
| HISPD    | = 000005   | #4-217 7-454                                                  |
| HWP      | 001264 R   | #4-243 4-253 7-472                                            |
| LF.ENA   | = ***** GX | 9-632 9-645                                                   |
| LF.MFL   | = ***** GX | 9-628                                                         |
| LF.X2P   | = ***** GX | 9-602                                                         |
| LINID    | 000226 R   | #4-192 10-675                                                 |
| LINMSG   | 000216 R   | #4-191 6-383 6-404                                            |
| L\$CNT   | = ***** GX | 10-680                                                        |
| L\$NAM   | = ***** GX | 7-473 10-677                                                  |
| L\$TRI   | = ***** GX | 10-694                                                        |
| L\$UNT   | = ***** GX | 10-686                                                        |
| OFF      | 000742 R   | #4-212 4-213 9-631                                            |
| OFFL     | = 000003   | #4-213 9-630                                                  |
| ON       | 000740 R   | #4-210 4-211 9-635 9-648                                      |
| ONL      | = 000002   | #4-211 9-634 9-647                                            |
| OPTBLK   | 001446 R   | 4-271 #4-279 7-436                                            |
| OPTMSG   | 000413 R   | #4-195 4-262                                                  |
| OP.CNT   | = 000001   | #4-114 4-269 7-435                                            |
| OP.OPT   | = 000004   | #4-116 4-271 7-506                                            |
| OP.STA   | = 000010   | #4-117 4-275 7-435                                            |
| OP.UNT   | = 000002   | #4-115 4-270 7-435                                            |
| OP.X2P   | = 000020   | #4-118 4-272 4-273 4-274 9-615                                |
| PROTBL   | 001270 R   | #4-246 7-479 7-480                                            |
| RATETB   | 000000 R   | #4-123 4-140 7-462                                            |
| SCRACH   | 001252 R   | #4-242 7-463                                                  |
| SF.ENA   | = ***** GX | 9-643                                                         |
| STABLK   | 001456 R   | 4-275 #4-280 6-407 *9-596 *9-597 *9-598 *9-626 *9-627 *9-630  |
|          |            | *9-631 *9-634 *9-635 *9-647 *9-648                            |
| STAMSG   | 000631 R   | #4-199 4-266 6-406                                            |
| S\$LCIT  | = ***** GX | 9-617                                                         |
| S\$LF LG | = ***** GX | 9-602 9-628 9-632                                             |
| TBLN     | = 000034   | #4-140                                                        |
| TBL1     | 001344 R   | #4-260 6-384                                                  |
| TBL2     | 001364 R   | #4-269 6-385                                                  |
| UBE      | 000775 R   | 4-150 #4-224                                                  |
| UBF      | 001000 R   | 4-151 #4-225                                                  |
| UBH      | 001003 R   | 4-152 #4-226                                                  |
| UBJ      | 001006 R   | 4-153 #4-227                                                  |
| UBK      | 001011 R   | 4-154 #4-228                                                  |
| UBL      | 001014 R   | 4-155 #4-229                                                  |
| UBM      | 001017 R   | 4-156 #4-230                                                  |
| UBN      | 001022 R   | 4-157 #4-231                                                  |
| UBP      | 001025 R   | 4-158 #4-232                                                  |
| UBR      | 001030 R   | 4-159 #4-233                                                  |
| UBS      | 001033 R   | 4-160 #4-234                                                  |
| UBT      | 001036 R   | 4-161 #4-235                                                  |
| UNTBK    | 001432 R   | 4-270 #4-278 *7-445 *7-446 *7-449 *7-450 *7-454 *7-455 *7-464 |
|          |            | 7-467 *7-479 *7-480 *7-491 *7-492                             |
| UNTMMSG  | 000337 R   | #4-194 4-261                                                  |
| URM      | 001041 R   | #4-236 8-550                                                  |
| URMTBL   | 000036 R   | #4-145 8-559                                                  |

.SBTTL PRTEVT - PRINT EVENT CHARACTERISTICS

PRTEVT - PRINT EVENT CHARACTERISTICS

INPUTS:

R5 - ADDRESS OF EVENT TEMPLATE BLOCK

OUTPUTS:

THE EVENT MESSAGE IS DISPLAYED

```

312
313
314
315
316
317
318
319
320
321
322
323
324
325 001014 105067 177272 PRTEVT: CLRB BITNUM ; START WITH EVENT TYPE 0
326 001020 142767 000003 177265 BICB #F.STRT!F.PREV.FLAGS ; CLEAR FLAGS WORD
327 001026 132767 000004 177257 BITB #F.NINF,FLAGS ; ANY INFORMATION PRINTED YET?
328 001034 001407 BEQ 10$; BR IF YES
329 001036 PRINT #EVTMSG ; PRINT EVENT DISPLAY HEADER
330 001046 142767 000004 177237 BICB #F.NINF,FLAGS ; CLEAR NO INFORMATION BIT
331 001054 012700 000000G 10$: MOV # $BUFF,R0 ; POINT TO BUFFER
332 001060 112720 000011 MOV #TAB,(R0)+ ; STORE TAB IN BUFFER
333 001064 016501 000004 MOV C.CLS(R5),R1 ; GET EVENT CLASS (BITS 6-15)
334 000006 .REPT 6 ;
335 ASR R1 ; MOVE EVENT CLASS TO BITS 0-9
336 .ENDR
337 001104 005002 CLR R2 ; SUPPRESS ZEROES
338 001106 004767 CALL $CBDMG ; CONVERT TO ASCII
339 001112 112720 000056 MOV #PERIOD,(R0)+ ; STORE PERIOD AFTER EVENT CLASS
340 001116 126727 177170 000100 20$: CMPB BITNUM,#MSKLEN ; ALL BITS PROCESSED?
341 001124 103065 BHS 80$; BR IF YES
342 001126 004767 000222 CALL CHKBIT ; TEST BIT TO SEE IF IT IS ON
343 001132 001455 BEQ 60$; BR IF NOT ON
344 001134 132767 000001 177151 BITB #F.STRT,FLAGS ; HAS FORMATTING BEGUN ALREADY?
345 001142 001016 BNE 30$; BR IF YES
346 001144 116701 177142 MOVBITNUM,R1 ; GET EVENT TYPE NUMBER
347 001150 005002 CLR R2 ; ZERO SUPPRESSION
348 001152 004767 000000G CALL $CBDMG ; CONVERT TO ASCII
349 001156 152767 000001 177127 BISB #F.STRT,FLAGS ; INDICATE FORMATTING HAS BEGUN
350 001164 152767 000002 177121 BISB #F.PREV,FLAGS ; INDICATE THIS EVENT TYPE IS SET
351 001172 105067 177116 CLRB PREVNT ; INDICATE FIRST NUMBER IN RANGE
352 001176 000435 BR 70$; CONTINUE
353
354 001200 132767 000002 177105 30$: BITB #F.PREV,FLAGS ; WAS PREVIOUS NUMBER SET?
355 001206 001015 BNE 40$; BR IF YES
356 001210 112720 000054 MOV #COMMA,(R0)+ ; SEPARATE WITH A COMMA
357 001214 116701 177072 MOVBITNUM,R1 ; GET EVENT TYPE NUMBER TO DISPLAY
358 001220 105067 177070 CLRB PREVNT ; INDICATE VALUE ALREADY DISPLAYED
359 001224 005002 CLR R2 ; ZERO SUPPRESSION
360 001226 004767 000000G CALL $CBDMG ; CONVERT TO ASCII
361 001232 152767 000002 177053 BISB #F.PREV,FLAGS ; INDICATE THIS EVENT TYPE IS SET
362 001240 000414 BR 70$; CONTINUE
363
364 001242 126027 177777 000055 40$: CMPB -1(R0),#DASH ; IS DASH ALREADY INSERTED?
365 001250 001402 BEQ 50$; BR IF YES
366 001252 112720 000055 MOV #DASH,(R0)+ ; STORE DASH
367 001256 116767 177030 50$: MOVBITNUM,PREVNT ; SAVE EVENT TYPE NUMBER
368 001264 000402 BR 70$; CONTINUE

```

```

153 .SBTTL LOCAL DATA
154
155 :****
156 : LOCAL DATA
157 :****
158 000000 HDRPRT: .BLKW 1 ; HEADER PRINTED FLAG
159 000002 LINHDR: .BLKW 1 ; ADDRESS OF LINE HEADER
160 000004 OBUF: .BLKW 1 ; OUTPUT BUFFER POINTER
161 000006 KNOWN: .BLKW 1 ; KNOWN DESTINATIONS FLAG
162 000010 DTYPE: .BLKW 1 ; DESTINATION TYPE (X25 OR X29)
163 000012 OPTION: .BLKW 1 ; OPTIONS
164 000014 SCRBLK: .BLKW 10 ; SCRATCH BLOCK
165 000034 LINBUF: ;
166 000034 CHNBUF: .BLKB 60. ; OUTPUT BUFFER FOR CHANNEL LIST
167 CHNBND = .
168
169 :
170 : ERROR TEXT
171 :
172 :
173 000130 002 103 106 ERRREQ: .ASCIZ <EX$ERR>/CFE -- Unknown information type/
174
175 :
176 : FORMAT STRINGS
177 000171 040 040 040 NOINF: .ASCIZ / No information/
178 000213 012 115 157 HDRMSG: .ASCIZ <12>/Module %VA %VA as of %Y %3Z/
179 000250 054 040 045 FMTRN1: .ASCIZ / %M/
180 000255 054 040 045 FMTRNM: .ASCIZ / %M-%M/
181 000265 045 115 040 FMTRSN: .ASCIZ /%M - %M/
182 000275 045 115 000 FMTRSR1: .ASCIZ /%M/
183 000300 045 062 122 FMTR2R5: .ASCIZ /%2R/
184
185 :
186 : X25 PROTOCOL CHARACTERISTICS
187 :
188 000304 BEGTXT PRCH
189 000304 TEXT 2,, <%N Default data = %M, Maximum data = %M>
190 000355 TEXT 2,, <%N Default window = %M, Maximum window = %M>
191 000430 TEXT 2,, <%N Reset timer = %M, Maximum resets = %M>
192 000500 TEXT 2,, <%N Restart timer = %M, Maximum restarts = %M>
193 000554 TEXT 2,, <%N Clear timer = %M, Maximum clears = %M>
194 000624 TEXT 3,, <%N Network name = %2R, Call timer = %M>
195 000672 ENDTXT
196
197 :
198 : X25-PROTOCOL DTE/KNOWN DTES
199 :
200 000724 BEGTXT DTE
201 000724 TEXT 5,, <%NDTE = %5R>
202 000740 TEXT 3,, <%N Counter timer = %M, Line = %VA>
203 001001 TEXT 2,, <%N State = %1R, Maximum circuits = %M>
204 001046 ENDTXT
205
206 001064 040 040 103 CHNHDR: .ASCIZ / Channels =/
207 001101 040 040 040 CHNBLK: .ASCIZ / /
208
209 :
210 : X25-PROTOCOL GROUP/KNOWN GROUPS
211 :
212 001116 BEGTXT CUG

```

SHOMOD - LIST MODULE CHARACTERI MACRO V05.03b Saturday 29-Jun-85 05:26 Page 16  
XACCHA - SHOW X25-ACCESS MODULE CHARACTERISTICS

```
657 .SBTTL XACCHA - SHOW X25-ACCESS MODULE CHARACTERISTICS
658 ;+
659 ;** XACCHA - SHOW X25-ACCESS MODULE CHARACTERISTICS
660 ;
661 ; INPUT $OPRND = REQUEST TYPE
662 ;
663 ; OUTPUT THE CURRENT IMPLEMENTATION OF PSI HAS NOT IMPLEMENTED
664 ; ANY MODULE X25-ACCESS PARAMETERS OTHER THAN THE
665 ; REMOTE DESTINATION LIST. THIS ROUTINE WILL ALWAYS
666 ; RETURN 'NO INFORMATION'.
667 ; -
668
669 003634 XACCHA:
670 003634 000207 RETURN
```

SHOMOD - LIST MODULE CHARACTERI MACRO V05.03b Saturday 29-Jun-85 05:26 Page 17  
SHOXAD - SHOW X25-ACCESS MODULE DESTINATIONS



SHOMOD      CREATED BY    MACRO    ON 29-JUN-85 AT 05:27      PAGE 4      H 5

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE      | REFERENCES |         |         |        |        |        |        |        |
|---------|------------|------------|---------|---------|--------|--------|--------|--------|--------|
| XA.DST  | = ***** GX | 11-527     | 11-535  |         |        |        |        |        |        |
| XA.KND  | = ***** GX | 11-527     | 11-536  | 17-687  |        |        |        |        |        |
| XPR     | = 001646 R | #5-242     | 5-242   | 6-281   |        |        |        |        |        |
| XPRCHA  | = 002156 R | 6-290      | #7-308  |         |        |        |        |        |        |
| XPRCUG  | = 002774 R | 6-293      | 6-294   | #10-472 |        |        |        |        |        |
| XPRDTB  | = 002104 R | 6-278      | #6-290  |         |        |        |        |        |        |
| XPRDTE  | = 002326 R | 6-291      | 6-292   | #8-354  |        |        |        |        |        |
| XPRL    | = 000014   | #5-242     | 6-280   |         |        |        |        |        |        |
| XP.CUG  | = ***** GX | 6-283      | 6-293   |         |        |        |        |        |        |
| XP.DTE  | = ***** GX | 6-283      | 6-291   |         |        |        |        |        |        |
| XP.KND  | = ***** GX | 6-283      | 6-292   | 8-360   |        |        |        |        |        |
| XP.KNG  | = ***** GX | 6-283      | 6-294   | 10-485  |        |        |        |        |        |
| XSV     | = 001662 R | #5-243     | 5-243   | 12-557  |        |        |        |        |        |
| XSVCBK  | = 000000 R | #5-218     | 13-584  | 15-650  |        |        |        |        |        |
| XSVCB1  | = 000000 R | #5-218     | 5-218   |         |        |        |        |        |        |
| XSVCF1  | = 001260 R | #5-218     | 13-596  | 15-653  |        |        |        |        |        |
| XSVCHA  | = 003400 R | 12-567     | #13-583 |         |        |        |        |        |        |
| XSVCM1  | = 001202 R | #5-217     | 5-218   |         |        |        |        |        |        |
| XSVDTB  | = 003346 R | 12-554     | #12-567 |         |        |        |        |        |        |
| XSVL    | = 000012   | #5-243     | 12-556  |         |        |        |        |        |        |
| XS.DST  | = ***** GX | 12-559     | 12-568  |         |        |        |        |        |        |
| XS.KND  | = ***** GX | 12-559     | 12-569  | 21-834  |        |        |        |        |        |
| X\$3CT  | = ***** GX | 7-336      |         |         |        |        |        |        |        |
| X\$3DFP | = ***** GX | 7-315      |         |         |        |        |        |        |        |
| X\$3DW  | = ***** GX | 7-317      |         |         |        |        |        |        |        |
| X\$3KM  | = ***** GX | 7-331      |         |         |        |        |        |        |        |
| X\$3KT  | = ***** GX | 7-329      |         |         |        |        |        |        |        |
| X\$3MW  | = ***** GX | 7-319      |         |         |        |        |        |        |        |
| X\$3MXP | = ***** GX | 7-316      |         |         |        |        |        |        |        |
| X\$3RM  | = ***** GX | 7-323      |         |         |        |        |        |        |        |
| X\$3RT  | = ***** GX | 7-321      |         |         |        |        |        |        |        |
| X\$3SM  | = ***** GX | 7-327      |         |         |        |        |        |        |        |
| X\$3ST  | = ***** GX | 7-325      |         |         |        |        |        |        |        |
| X\$9CT  | = ***** GX | 15-652     |         |         |        |        |        |        |        |
| X\$9MXC | = ***** GX | 15-651     |         |         |        |        |        |        |        |
| X29     | = 001706 R | #5-245     | 5-245   | 14-618  |        |        |        |        |        |
| X29CHA  | = 003566 R | 14-628     | #15-644 |         |        |        |        |        |        |
| X29DTB  | = 003534 R | 14-615     | #14-628 |         |        |        |        |        |        |
| X29L    | = 000012   | #5-245     | 14-617  |         |        |        |        |        |        |
| X9.DST  | = ***** GX | 14-620     | 14-629  |         |        |        |        |        |        |
| X9.KND  | = ***** GX | 14-620     | 14-630  |         |        |        |        |        |        |
| \$BUFF  | = ***** GX | 19-767     | 19-769  |         |        |        |        |        |        |
| \$CBTA  | = ***** GX | 20-794     | 20-800  | 20-808  |        |        |        |        |        |
| \$CFERR | = ***** GX | 18-737     |         |         |        |        |        |        |        |
| \$CUG   | = ***** GX | 10-487     | 10-489  |         |        |        |        |        |        |
| \$C5TA  | = ***** GX | 20-790     |         |         |        |        |        |        |        |
| \$DST   | = ***** GX | 17-689     | 17-691  | 21-836  | 21-838 |        |        |        |        |
| \$DTE   | = ***** GX | 8-362      |         |         |        |        |        |        |        |
| \$EDMSG | = ***** GX | 9-439      | 19-768  | 21-872  |        |        |        |        |        |
| \$FLAGS | = ***** GX | *8-389     | *8-392  | 20-795  | 20-801 | 20-803 |        |        |        |
| \$HDBK  | = ***** GX | 6-279      | 11-523  | 12-555  | 14-616 | 19-759 |        |        |        |
| \$OPRND | = ***** GX | 6-282      | 8-360   | 10-485  | 11-526 | 12-558 | 14-619 | 17-687 | 21-834 |

\*\*FILE\*\*ID\*\*SHOREM

H 6

```

SSSSSSSS HH HH 000000 RRRRRRRR EEEEEEEEE MM MM
SSSSSSSS HH HH 000000 RRRRRRRR EEEEEEEEE MM MM
SS HH HH 00 00 RR RR EE MMMM MMMM
SS HH HH 00 00 RR RR EE MMMM MMMM
SS HH HH 00 00 RR RR EE MM MM MM
SSSSSS HHHHHHHHH 00 00 RRRRRRRR EEEEEEEE MM MM
SSSSSS HHHHHHHHH 00 00 RRRRRRRR EEEEEEEE MM MM
SS HH HH 00 00 RR RR EE MM MM
SS HH HH 00 00 RR RR EE MM MM
SS HH HH 00 00 RR RR EE MM MM
SSSSSS HH HH 000000 RR RR EEEEEEEEE MM MM
SSSSSS HH HH 000000 RR RR EEEEEEEEE MM MM

```

....  
....  
....  
....

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSS TT
LL SSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLL SSSSSSS TT
LLLLLLLLL SSSSSSS TT

```

```

484 .SBTTL PUTFIL - OUTPUT A FILESPEC PARAMETER
485 ;+
486 ;** PUTFIL - OUTPUT A FILESPEC PARAMETER
487 ;
488 ; INPUTS: R0 = buffer pointer
489 ; R2 = address of file descriptor pointer
490 ;
491 ; OUTPUT: R0 is updated.
492 ; -
493
494 001560 PUTFIL: SAVRG R3 ; SAVE REGISTER
495 001562 004567 000006 JSR R5,20$; MOVE FILESPEC
496 001566 000005 .WORD F.DEV ;
497 001570 RESRG R3 ; RESTORE REGISTER
498 001572 000207 RETURN
499
500 ; MOVE A FIELD
501
502 001574 010201 20$: MOV R2,R1 ; COPY ADDRESS OF FILE DESCRIPTOR
503 001576 062501 ADD (R5)+,R1 ; POINT TO FIELD OF INTEREST
504 001600 116103 177777 MOVB -1(R1),R3 ; GET FIELD LENGTH
505 001604 001402 BEQ 40$; IF EQ, NO TEXT
506 001606 112120 30$: MOVB (R1)+,(R0)+ ; ELSE MOVE IT
507 001610 077302 SOB R3,30$; TO OUTPUT BUFFER
508 001612 000205 40$: RTS R5 ; DONE
509

```

```

115 000000 002 103 106 ERRPAR: .ASCIZ <EX$ERR>/CFE -- No partition definition/
116 000040 002 103 106 ERRBUF: .ASCIZ <EX$ERR>/CFE -- No buffer definition/
117 000075 012 123 171 HEADR: .ASCIZ <12>/System characteristics as of %Y %3Z/
118 000142 012 040 040 PARMSG: .ASCIZ <12>/ Pool name = %2R, Partition name = %2R/
119 000213 040 040 102 PAR2MS: .ASCIZ / Byte-area extension = %M, Location = %VA/
120 000266 124 157 160 TOP: .ASCII /Topdown/
121 000007 -TOP
122 000275 106 151 162 FIRST: .ASCII /Firstfit/
123 000010 -FIRST
124 000305 040 040 115 01MSG: .ASCIZ / Maximum control buffers = %M, Maximum small buffers = %M/
125 000400 040 040 115 02MSG: .ASCIZ / Maximum large buffers = %M, Large buffer size = %M/
126 000465 040 040 115 03MSG: .ASCIZ / Minimum receive buffers = %M/
127 .EVEN
128
129 000524 063147 047574 POL: .RAD50 /POOL../
130 000530 BUF: .BLKW 5 ; ARGUMENTS BUFFER
131 000542 000305' MSTBL: .WORD 01MSG ; MAX CONTROL BUFFERS, MAX SMALL BUFFERS
132 000544 000400' .WORD 02MSG ; MAX LARGE BUFFERS, LARGE BUFFER SIZE
133 000546 000465' .WORD 03MSG ; RECEIVE BUFFER MIN.
134 000550 000000 .WORD 0 ; END OF TABLE
135
136 .SBTTL $SHSYS - SHOW SYSTEM CHARACTERISTICS
137
138 ;+
139 ; *** - $SHSYS - SHOW SYSTEM CHARACTERISTICS
140
141 INPUT:
142 NONE.
143
144 OUTPUT:
145 THE SYSTEM CHARACTERISTICS ARE DISPLAYED.
146
147 ; -
148
149 000552 $SHSYS:
150 000556 012700 000000G MOV # $BUFF,R0 ; POINT TO OUTPUT BUFFER
151 000562 012701 000075' MOV # $HEADR,R1 ; POINT TO FORMAT STRING
152 000566 012702 000000G MOV # $GTBUF,R2 ; POINT TO ARGUMENT BLOCK
153 000572 004767 000000G CALL $EDMSG ; FORMAT THE HEADER
154 000602 012705 000000G PRINT $ $BUFF ; PRINT THE HEADER
155 000606 010504 MOV # $TMLST,R5 ; GET START OF TEMPLATE LIST
156 MOV R5,R4 ; ..TWICE
157
158 ; SEARCH FOR THE PARTITION TEMPLATE
159
160 000610 011505 10$: MOV (R5),R5 ; GET NEXT TEMPLATE
161 000612 001002 BNE 12$; IF NE, MORE
162 000614 000167 000300 JMP 101$; ELSE, END OF TEMPLATES
163 000620 122765 000000G 12$: CMPB # $CS.PAR,C.STS(R5) ; IS THIS A PARTITION TEMPLATE ?
164 000626 001370 BNE 10$; IF NE, NO - KEEP LOOKING
165 000630 010502 MOV R5,R2 ; POINT R2 AT ARGUMENT BLOCK
166 000632 062702 000004 ADD # $C.NAM0,R2 ;
167 000636 005712 TST (R2) ; IS THE NAME DEFINED ?
168 000640 001010 BNE 5$; IF NE, YES
169 000642 012702 000524' MOV $POL,R2 ; ELSE, USE DEFAULT
170 000646 016567 000010 177654 MOV C.NAM1(R5),POL+4 ; STORE PARTITION NAME
171 000654 016567 000012 177650 MOV C.NAM1+2(R5),POL+6 ;
172 000662 012700 000000G 5$: MOV # $BUFF,R0 ; POINT AT OUTPUT BUFFER

```

234  
 235  
 236  
 237  
 238  
 239  
 240  
 241  
 242  
 243  
 244  
 245  
 246  
 247  
 248  
 249  
 250  
 251  
 252  
 253  
 254  
 255  
 256  
 257  
 258  
 259  
 260  
 261  
 262  
 263

000134  
 000134 016700 177640  
 000140 016701 000000G  
 000144 010160 000004  
 000150 010167 000000G  
 000154 020167 177622  
 000160 103403  
 000162 020167 177616  
 000166 101402  
 000170 062716 000002  
 000174 000207

```

.SBTTL SETCBN - SET TO CHANGE NUMBER
+
*** - SETCBN - SET BUFFER INFO IN END OF TASK TEMPLATE
REASONABLE CHECKS ARE PERFORMED ON THE CURRENT VALUE. IF THE VALUE IS
NOT IN RANGE, THE C-BIT IS SET.

INPUT:
TEMP = TEMPLATE ADDRESS
$LIMITB = ADDRESS OF LIMITS TABLE
.PNUMB = CURRENT VALUE
.PNUMH = CURRENT VALUE (HIGH ORDER)

OUTPUT:
C-BIT = SUCCESS/FAILURE
VALUE IS STORED IN TEMPLATE.
-

SETCBN: MOV TEMP,R0 ; POINT R0 AT TEMPLATE
 MOV .PNUMB,R1 ; GET THE NUMBER OF RDB'S
 MOV R1,C.CBNM(R0) ; STORE THE CCB NUMBER
 MOV R1,$CCBNM ; SAVE THE NUMBER OF CCB'S FOR LATER
 CMP R1,LOCBNM ; IS THE VALUE IN RANGE ?
 BLO 10$; IF LO, NO .. REJECT
 CMP R1,HICBNM ; IS THE VALUE STILL IN RANGE ?
 BLOS 20$; IF LOS, YES .. OKAY
 ADD #2,(SP) ; REJECT THE TRANSITION (SETS C-BIT)
10$: ADD
20$: RETURN

```

TMPCHN - PARSE OUTGOING CHANNEL MACRO V05.03b Saturday 29-Jun-85 05:28 <sup>I 10</sup>  
Table of contents

|    |     |                                                   |
|----|-----|---------------------------------------------------|
| 4- | 57  | MACRO DEFINITIONS                                 |
| 5- | 126 | LOCAL DATA                                        |
| 6- | 136 | TPARS STATE TABLES                                |
| 7- | 164 | \$TMCHN - BUILD CHANNEL TEMPLATE                  |
| 8- | 202 | SETLO - SET LOW END OF CHANNEL RANGE IN TEMPLATE  |
| 8- | 203 | SETHI - SET HIGH END OF CHANNEL RANGE IN TEMPLATE |

.TITLE TMPCNT - PARSE CONTROLLER DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE CONTROLLER DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

TMPCUG - PARSE CLOSED USER GROU MACRO V05.03b Saturday 29-Jun-85 05:29 <sup>H 12</sup>  
Table of contents

|    |     |                               |
|----|-----|-------------------------------|
| 4- | 57  | MACRO DEFINITIONS             |
| 5- | 128 | LOCAL DATA                    |
| 6- | 153 | TPARS STATE TABLES            |
| 7- | 192 | \$TMPCUG - BUILD EUG TEMPLATE |
| 7- | 230 | SETNAM - SET CUG NAME         |
| 7- | 231 | SETCUG - SET GROUP NUMBER     |
| 7- | 232 | SETFLG - SET FLAGS BYTE       |



1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53

.TITLE TMPDDM - PARSE DEVICE PROCESS DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE DDM DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

\*\*FILE\*\*ID\*\*TMPDEC

```

TTTTTTTTTT MM MM PPPPPPPP DDDDDDDD EEEEEEEEE CCCCCCCC
TTTTTTTTTT MM MM PPPPPPPP DDDDDDDD EEEEEEEEE CCCCCCCC
 TT MMMM MMMM PP PP DD DD EE CC
 TT MMMM MMMM PP PP DD DD EE CC
 TT MM MM PP PP DD DD EE CC
 TT MM MM PP PP DD DD EE CC
 TT MM MM PPPPPPPP DD DD EEEEEEEE CC
 TT MM MM PPPPPPPP DD DD EEEEEEEE CC
 TT MM MM PP DD DD EE CC
 TT MM MM PP DD DD EE CC
 TT MM MM PP DD DD EE CC
 TT MM MM PP DD DD EE CC
 TT MM MM PP DDDDDDDD EEEEEEEEE CCCCCCCC
 TT MM MM PP DDDDDDDD EEEEEEEEE CCCCCCCC

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

325  
 326  
 327  
 328  
 329  
 330  
 331  
 332  
 333  
 334  
 335  
 336  
 337  
 338  
 339  
 340  
 341  
 342  
 343  
 344  
 345  
 346  
 347  
 348  
 349

```

 .SBTTL SETDWF - DELAY WEIGHT
 *
 *** - SETDEW - SETUP DELAY WEIGHT
 THIS ACTION ROUTINE SETS UP THE DELAY WEIGHT FOR THE EXECUTOR
 INPUT:
 .PNUMB = DELAY WEIGHT
 .PNUMH = (HIGH ORDER)
 OUTPUT:
 THE DELAY WEIGHT IS STORED IN C.DELW OF THE TEMPLATE
 -
 SETDEW:
 MOV TEMP,RO ; GET START OF TEMPLATE
 MOVB .PNUMB,C.DELW(RO) ; STORE DELAY WEIGHT
 TST .PNUMH ; IS IT ONLY A WORD VALUE ?
 BNE 10$; IF NE, NO - REJECT
 TSTB .PNUMB+1 ; IS IT A BYTE VALUE ?
 BEQ 20$; IF EQ, YES - OKAY
 10$: ADD #2,(SP) ; ELSE, REJECT TRANSITION
 20$: RETURN

```

```

000344 177430
000344 016700 177430
000350 116760 000000G 000017
000356 005767 000000G
000362 001003
000364 105767 000001G
000370 001402
000372 062716 000002
000376 000207

```

```

92 ;****
93 ; INITIALIZE TABLE GENERATION
94 ;****
95
96 000004 ISTAT$ DLCSTB,DLCKTB
97
98 000004 STATES$
99 000004 TRANS$ $RAD50,,SEINAM
100
101 000004 STATES$
102 000004 TRANS$ '<','>'
103
104 000004 STATES$
105 000004 TRANS$ '!ZF'
106
107 000004 STATES$
108 000004 TRANS$ '"DLC"',,,ZF.DLC,FLAG
109
110 000004 STATES$ BITS
111 000004 TRANS$ '! ,BITO
112 000004 TRANS$ $LAMDA,PRIOR,SETFLG
113
114 000004 STATES$ BITO
115 000004 TRANS$ '!ZF'
116
117 000004 STATES$
118 000004 TRANS$ '"TIM"',BITS,,ZF.TIM,FLAG
119 000004 TRANS$ '"MFL"',BITS,,ZF.MFL,FLAG
120 000004 TRANS$ '"COU"',BITS,,ZF.COU,FLAG
121 000004 TRANS$ '"MAN"',BITS,,ZF.MAN,FLAG
122
123 000004 STATES$ PRIOR
124 000004 TRANS$ '<','>'
125
126 000004 STATES$
127 000004 TRANS$ $NUMBR,,SETPRI
128
129 000004 STATES$
130 000004 TRANS$ $EOS,$EXIT
131 000004 TRANS$ '<','>'
132
133 000004 STATES$
134 000004 TRANS$ $NUMBR,NCT,SETNLT
135 000004 TRANS$ $LAMDA
136
137 000004 STATES$ NCT
138 000004 TRANS$ $EOS,$EXIT
139 000004 TRANS$ '<','>'
140
141 000004 STATES$
142 000004 TRANS$ $NUMBR,$EXIT,SETNCT ; NUMBER OF CONTROLLER TABLES
143
144 000004 STATES$ ZF
145 000004 TRANS$ '"ZF"'
146
147 000004 STATES$
148 000004 TRANS$ '.,$EXIT

```

## SYMBOL CROSS REFERENCE

CREF 04.00

SYMBOL VALUE REFERENCES

|         |             |        |        |        |        |        |        |        |        |
|---------|-------------|--------|--------|--------|--------|--------|--------|--------|--------|
| URMVAL  | = 001070 R  | #4-237 | 8-541  |        |        |        |        |        |        |
| WC.CNT  | = ***** GX  | 5-327  | 5-330  | 5-347  |        |        |        |        |        |
| WC.NAM  | = ***** GX  | 5-299  |        |        |        |        |        |        |        |
| WC.UNT  | = ***** GX  | 5-354  |        |        |        |        |        |        |        |
| X\$2BSZ | = ***** GX  | 9-618  |        |        |        |        |        |        |        |
| X\$2HBT | = ***** GX  | 9-616  |        |        |        |        |        |        |        |
| X\$2LIN | = ***** GX  | 9-609  | 9-611  | 9-613  |        |        |        |        |        |
| X\$2RTC | = ***** GX  | 9-620  |        |        |        |        |        |        |        |
| X\$2RTT | = ***** GX  | 9-621  |        |        |        |        |        |        |        |
| X\$2WSZ | = ***** GX  | 9-619  |        |        |        |        |        |        |        |
| X2PBLK  | = 001466 R  | 4-272  | #4-281 | *9-616 | *9-617 |        |        |        |        |
| X2PBL2  | = 001500 R  | 4-273  | #4-282 | *9-618 | *9-619 |        |        |        |        |
| X2PBL3  | = 001504 R  | 4-274  | #4-283 | *9-620 | *9-621 |        |        |        |        |
| X2PLIN  | = 000710 R  | #4-202 | 9-623  |        |        |        |        |        |        |
| X2PMSG  | = 000422 R  | #4-196 | 4-263  |        |        |        |        |        |        |
| X2PMS2  | = 000474 R  | #4-197 | 4-264  |        |        |        |        |        |        |
| X2PMS3  | = 000545 R  | #4-198 | 4-265  |        |        |        |        |        |        |
| \$BUFF  | = ***** GX  | 5-302  | 5-303  | 6-387  | 6-394  | 6-405  | 6-409  |        |        |
| \$CBDMG | = ***** GX  | 7-466  | 7-501  |        |        |        |        |        |        |
| \$CBOMG | = ***** GX  | 7-515  |        |        |        |        |        |        |        |
| \$CBTA  | = ***** GX  | 10-682 | 10-688 | 10-696 |        |        |        |        |        |
| \$CFERR | = ***** GX  | 6-417  | 9-625  |        |        |        |        |        |        |
| \$CNTNM | = ***** GX  | 5-332  | *5-334 |        |        |        |        |        |        |
| \$CSTA  | = ***** GX  | 7-474  | 10-678 |        |        |        |        |        |        |
| \$DVNAM | = ***** GX  | *5-318 | 7-475  | 9-609  | 9-611  | 9-613  | 10-676 |        |        |
| \$EDMSG | = ***** GX  | 5-304  | 6-393  | 6-408  |        |        |        |        |        |
| \$FLAGS | = ***** GX  | *5-307 | 5-352  | 6-396  | 6-400  | *6-402 | *6-412 | 9-636  | 9-638  |
|         |             | 9-654  | 10-683 | 10-689 | 10-691 |        |        |        | 9-652  |
| \$FNPRO | = ***** GX  | 5-311  |        |        |        |        |        |        |        |
| \$FNST  | = ***** GX  | 9-590  |        |        |        |        |        |        |        |
| \$FNSTA | = ***** GX  | 9-640  |        |        |        |        |        |        |        |
| \$GTBUF | = ***** GX  | 5-303  |        |        |        |        |        |        |        |
| \$PRINT | = ***** GX  | 5-305  | 6-383  | 6-394  | 6-404  | 6-409  |        |        |        |
| \$SHLIN | = 001510 RG | #5-297 |        |        |        |        |        |        |        |
| \$TMLST | = ***** GX  | 5-306  | 5-319  | 9-604  |        |        |        |        |        |
| \$UNTNM | = ***** GX  | 5-356  | *5-358 |        |        |        |        |        |        |
| \$WILD  | = ***** GX  | 5-299  | 5-327  | 5-330  | 5-347  | 5-354  |        |        |        |
| \$\$\$  | = 000015    | #4-246 | 4-246  | #4-248 | 4-248  | #4-249 | 4-249  | #4-250 | 4-250  |
|         |             | 4-251  | #4-252 | 4-252  | #4-255 | 4-255  | #4-256 | 4-256  | #4-257 |
| \$\$\$1 | = 000070 R  | #4-246 | 4-246  | #4-248 | 4-248  | #4-249 | 4-249  | #4-250 | 4-250  |
|         |             | 4-251  | #4-252 | 4-252  | #4-255 | 4-255  | #4-256 | 4-256  | #4-257 |

|     |        |        |        |       |        |         |  |                                            |
|-----|--------|--------|--------|-------|--------|---------|--|--------------------------------------------|
| 369 |        |        |        |       |        |         |  |                                            |
| 370 | 001266 | 004767 | 000026 | 60\$: | CALL   | CKPREV  |  | ; CHECK FOR DISPLAY OF PREVIOUS EVENT TYPE |
| 371 | 001272 | 105267 | 177014 | 70\$: | INCB   | BITNUM  |  | ; UPDATE BIT NUMBER                        |
| 372 | 001276 | 000707 |        |       | BR     | 20\$    |  |                                            |
| 373 |        |        |        |       |        |         |  |                                            |
| 374 | 001300 | 004767 | 000014 | 80\$: | CALL   | CKPREV  |  | ; SEE IF LAST EVENT TYPE MUST BE DISPLAYED |
| 375 | 001304 | 105020 |        |       | CLRB   | (RO)+   |  | ; CREATE ASCII STRING                      |
| 376 | 001306 |        |        |       | PRINT  | #\$BUFF |  | ; PRINT EVENT MESSAGE                      |
| 377 | 001316 | 000207 |        |       | RETURN |         |  |                                            |

```

LOCAL DATA
210 001116 TEXT 2,,<%NGroup = %2R>
211 001134 TEXT 8,,< Number = %M, DTE = %5R%VA>
212 001170 ENDTXT
213
214 ; X25-SERVER CHARACTERISTICS
215 ;
216 001202 BEGTXT XSVC
217 001202 TEXT 2,,<%N Maximum circuits = %M, Counter timer = %M>
218 001260 ENDTXT
219
220 ; X25/X29 SERVER DESTINATION/KNOWN DESTINATIONS
221 ;
222 001266 BEGTXT DST
223 001266 TEXT 2,,<%NDestination = %2R>
224 001312 TEXT 3,,< Priority = %M, Object = %VA>
225 001350 TEXT 5,DS.NUM,< Number = %5R>
226 001367 TEXT 2,DS.CUG,< Group = %2R>
227 001405 TEXT 1,DS.CLO,< Low subaddress = %M>
228 001433 TEXT 1,DS.CHI,< High subaddress = %M>
229 001462 TEXT 13,DS.MSK,< Call mask = %11R>
230 001505 TEXT 13,DS.VAL,< Call value = %11R>
231 001531 ENDTXT
232 DS.OPT = DS.NUM!DS.CUG!DS.CLO!DS.CHI!DS.MSK!DS.VAL
233
234 ; X25-ACCESS DESTINATION/KNOWN DESTINATIONS
235 ;
236 001574 BEGTXT XAD
237 001574 TEXT 2,,<%NDestination = %2R%N Number = %5R>
238 001640 ENDTXT
239
240 ; REPLACEMENT STRINGS
241 ;
242 001646 STRING XPR,<X25-Protocol>
243 001662 STRING XSV,<X25-Server>
244 001674 STRING XAC,<X25-Access>
245 001706 STRING X29,<X29-Server>
246 001720 STRING CHAR,<characteristics>
247 001737 STRING KDST,<known destinations>
248 001761 STRING KDTE,<known DTEs>
249 001773 STRING KCUG,<known groups>
250 002007 STRING TYPB,<, Bilateral>
251 002022 STRING BLANK,<>
252
253 .EVEN
254 .LIST BEX
255
256 ; SPECIAL X29 DESTINATIONS
257 ;
258 ; *** THERE IS A COPY OF THIS TABLE IN PURMOD ***
259 002024 115443 115447 003310 SPCDST: .RAD50 /X25/ /X29ACP/
260 002032 115447 050712 000000 .RAD50 /X29/ /MCR /
261 002040 000000 .WORD 0
262

```

```

672 .SBTTL SHOXAD - SHOW X25-ACCESS MODULE DESTINATIONS
673 ;+
674 ;** SHOXAD - SHOW X25-ACCESS MODULE DESTINATIONS
675 ;
676 ; INPUTS $OPRND = REQUEST TYPE
677 ;
678 ; OUTPUT THE X25-ACCESS REMOTE DESTINATIONS ARE DISPLAYED.
679 ; -
680
681 003636 SHOXAD:
682 003636 012705 000000G MOV #STMLST,R5 ; POINT TO START OF TEMPLATES
683 003642 011505 10$: MOV (R5),R5 ; NEXT TEMPLATE
684 003644 001440 BEQ 30$; IF EQ, END OF LIST
685 003646 122765 000000G 000000G CMPB #CS.RDT,C.STS(R5) ; IS THIS A REMOTE DTE DESCRIPTOR ?
686 003654 001372 BNE 10$; BR IF NO
687 003656 032767 000000G 000000G BIT #XA.KND,$OPRND ; KNOWN DESTINATIONS ?
688 003664 001010 BNE 20$; BR IF YES
689 003666 026765 000000G 000000G CMP DST,RDNAM(R5) ; COMPARE NAMES
690 003674 001362 BNE 10$; BR IF WRONG DESTINATION
691 003676 026765 000002G 000002G CMP $DST+2,R$DNAM+2(R5) ; COMPARE NAMES
692 003704 001356 BNE 10$; BR IF WRONG DESTINATION
693 003706 012701 000000' 20$: MOV #XADBK,R1 ; POINT TO ARGUMENT BLOCK
694 003712 010502 MOV R5,R2 ; COPY POINTER
695 003714 022222 CMP (R2)+,(R2)+ ; POINT PAST LINK AND STATUS
696 000007 .REPT 7
697 MOV (R2)+,(R1)+ ; COPY DATA
698 .ENDR
699 003734 012703 001640' MOV #XADFT,R3 ; POINT TO FORMATTING TABLE
700 003740 004767 000066 PRINT ; SHOW INFORMATION
701 003744 000736 BR 10$; CONTINUE
702 003746 000207 30$: RETURN

```



## SYMBOL CROSS REFERENCE

CREF 04.00

| SYMBOL | VALUE | REFERENCES |
|--------|-------|------------|
|--------|-------|------------|

| SPRINT  | *****  | GX | 9-449   | 9-457  | 18-734 | 19-769 |        |        |        |        |        |  |
|---------|--------|----|---------|--------|--------|--------|--------|--------|--------|--------|--------|--|
| \$PSIPT | *****  | GX | 7-333   | 13-585 |        |        |        |        |        |        |        |  |
| \$SHXAC | 003202 | RG | #11-521 |        |        |        |        |        |        |        |        |  |
| \$SHXPR | 002042 | RG | #6-277  |        |        |        |        |        |        |        |        |  |
| \$SHXSV | 003276 | RG | #12-553 |        |        |        |        |        |        |        |        |  |
| \$SHX29 | 003464 | RG | #14-614 |        |        |        |        |        |        |        |        |  |
| \$TMLST | *****  | GX | 7-309   | 8-355  | 10-473 | 13-588 | 15-645 | 17-682 | 21-826 |        |        |  |
| \$BASE  | 000000 | R  | 5-194   | 5-194  | 5-202  | 5-202  | 5-212  | 5-212  | 5-218  | 5-218  | 5-231  |  |
| \$SOBIT | 000004 |    | 5-231   | 5-238  | 5-238  |        |        |        |        |        |        |  |
|         |        |    | #5-187  | 5-188  | 5-188  | #5-188 | 5-189  | 5-189  | #5-189 | 5-190  | 5-190  |  |
|         |        |    | #5-190  | 5-191  | 5-191  | #5-191 | 5-192  | 5-192  | #5-192 | 5-193  | 5-193  |  |
|         |        |    | #5-193  | #5-198 | 5-199  | #5-199 | #5-199 | 5-200  | #5-200 | #5-200 | 5-201  |  |
|         |        |    | 5-201   | #5-201 | #5-209 | 5-210  | 5-210  | #5-210 | 5-211  | 5-211  | #5-211 |  |
|         |        |    | #5-216  | 5-217  | 5-217  | #5-217 | #5-222 | 5-223  | 5-223  | #5-223 | 5-224  |  |
|         |        |    | 5-224   | #5-224 | 5-225  | 5-225  | 5-225  | #5-225 | 5-226  | 5-226  | 5-226  |  |
|         |        |    | #5-226  | 5-227  | 5-227  | 5-227  | #5-227 | 5-228  | 5-228  | 5-228  | #5-228 |  |
|         |        |    | 5-229   | 5-229  | 5-229  | #5-229 | 5-230  | 5-230  | 5-230  | #5-230 | #5-236 |  |
|         |        |    | 5-237   | 5-237  | #5-237 |        |        |        |        |        |        |  |
| \$TTX0  | 000001 |    | #5-187  | 5-188  | #5-188 | 5-188  | 5-189  | #5-189 | 5-189  | 5-190  | #5-190 |  |
|         |        |    | 5-190   | 5-191  | #5-191 | 5-191  | 5-192  | #5-192 | 5-192  | 5-193  | #5-193 |  |
|         |        |    | 5-193   | 5-194  | 5-194  | #5-198 | 5-199  | #5-199 | 5-200  | #5-200 | 5-200  |  |
|         |        |    | 5-200   | 5-201  | #5-201 | 5-201  | 5-202  | 5-202  | #5-209 | 5-210  | #5-210 |  |
|         |        |    | 5-210   | 5-211  | #5-211 | 5-211  | 5-212  | 5-212  | #5-216 | 5-217  | #5-217 |  |
|         |        |    | 5-217   | 5-218  | #5-218 | #5-222 | 5-223  | #5-223 | 5-223  | 5-224  | #5-224 |  |
|         |        |    | 5-224   | 5-225  | #5-225 | 5-225  | 5-226  | #5-226 | 5-226  | 5-227  | #5-227 |  |
|         |        |    | 5-227   | 5-228  | #5-228 | 5-228  | 5-229  | #5-229 | 5-229  | 5-230  | #5-230 |  |
| \$TTX1  | 000002 |    | 5-230   | 5-231  | 5-231  | #5-236 | 5-237  | #5-237 | 5-237  | 5-238  | 5-238  |  |
|         |        |    | #5-194  | 5-194  | 5-194  | #5-194 | 5-194  | 5-194  | #5-194 | 5-194  | 5-194  |  |
|         |        |    | #5-194  | 5-194  | 5-194  | #5-194 | 5-194  | 5-194  | #5-194 | 5-194  | 5-194  |  |
|         |        |    | #5-194  | #5-194 | 5-194  | 5-194  | #5-194 | 5-194  | #5-194 | #5-197 | 5-194  |  |
|         |        |    | 5-194   | 5-194  | 5-194  | 5-194  | #5-194 | 5-194  | 5-194  | #5-194 | 5-194  |  |
|         |        |    | 5-194   | #5-194 | #5-202 | 5-202  | 5-202  | #5-202 | 5-202  | 5-202  | #5-202 |  |
|         |        |    | 5-202   | 5-202  | 5-202  | #5-202 | 5-202  | 5-202  | #5-202 | 5-202  | 5-202  |  |
|         |        |    | #5-202  | 5-202  | 5-202  | #5-202 | 5-212  | #5-212 | 5-212  | #5-212 | 5-212  |  |
|         |        |    | 5-212   | #5-212 | #5-212 | 5-212  | 5-212  | #5-212 | 5-212  | 5-212  | #5-212 |  |
|         |        |    | #5-218  | 5-218  | 5-218  | #5-218 | #5-218 | 5-218  | 5-218  | #5-218 | 5-218  |  |
|         |        |    | 5-231   | 5-231  | #5-231 | 5-231  | 5-231  | #5-231 | 5-231  | 5-231  | #5-231 |  |
|         |        |    | 5-231   | 5-231  | 5-231  | 5-231  | 5-231  | 5-231  | 5-231  | 5-231  | #5-231 |  |
|         |        |    | 5-231   | 5-231  | #5-231 |        |        |        |        |        |        |  |

SHOREM - SHOW REMOTE NODE CHARA MACRO V05.03b Saturday 29-Jun-85 05:27 <sup>16</sup>  
Table of contents

|     |     |                                                   |
|-----|-----|---------------------------------------------------|
| 4-  | 240 | SSHREM - SHOW REMOTE CHARACTERISTICS              |
| 7-  | 379 | PUTADR - PUT A PHASE 4 ADDRESS                    |
| 8-  | 402 | GETDLL - GET DOWNLINE LOAD INFO                   |
| 9-  | 484 | PUTFIL - OUTPUT A FILESPEC PARAMETER              |
| 10- | 511 | PUTxxx - MISCELLANEOUS OUTPUT FORMATTING ROUTINES |

```

511 .SBTTL PUTxxx - MISCELLANEDUS OUTPUT FORMATTING ROUTINES
512
513
514 ;+
515 ;** PUTNDD - OUTPUT NODE ADDRESS
516 ;
517 ; INPUTS: R0 = BUFFER PDINTER
518 ; R2 = OFFSET OF PARAMETER
519 ;
520 ; OUTPUTS: R0 IS UPDATED.
521 ; -
522
523 001614 010201 PUTNDD: MOV R2,R1 ; COPY OFFSET
524 001616 066701 176522 ADD SERPT,R1 ; GET ADDRESS OF DATA
525 001622 011101 MOV (R1),R1 ; GET NODE ADDRESS
526 001624 010146 MOV R1,-(SP) ; PUSH NODE ADDRESS
527 001626 010146 MOV R1,-(SP) ; ONCE MORE FOR GOOD LUCK
528 001630 042701 001777 BIC #1777,R1 ; ISOLATE AREA BITS
529 001634 000301 SWAB R1 ; SHIFT (AND CLEAR CARRY)
530 001636 006201 ASR R1 ; ...
531 001640 006201 ASR R1 ; ...
532 001642 005701 TST R1 ; AREA 0?
533 001644 001405 BEQ 10$, ; IF EQ, NO AREA SHOWN
534 001646 005002 CLR R2 ; SUPPRESS ZEROES
535 001650 004767 000000G CALL $CDBMG ; CONVERT IT
536 001654 112720 000056 MOV #<'>,(R0)+ ; INSERT DELIMITER
537 001660 012601 MOV (SP)+,R1 ; RECOVER NODE ADDRESS
538 001662 042701 176000 BIC #^C1777,R1 ; CLEAR AREA BITS
539 001666 005002 CLR R2 ; NO ZEROES
540 001670 004767 000000G CALL $CDBMG ; CONVERT IT
541 001674 012601 MOV (SP)+,R1 ; RECOVER NODE ADDRESS AGAIN
542 001676 012702 000000G MOV #$TMLST,R2 ; POINT TO NODE LIST
543 001702 020167 000000G CMP R1,$NTADD ; EXECUTOR ADDRESS ?
544 001706 001005 BNE 20$, ; NO
545 001710 016746 000002G MOV $NTNAM+2,-(SP) ; GET EXEC NAME
546 001714 016701 000000G MOV $NTNAM,R1 ;
547 001720 000415 BR 25$, ; JOIN COMMON CODE
548
549 001722 011202 20$: MOV (R2),R2 ; NEXT
550 001724 001426 BEQ 40$, ; IF EQ, NO MORE
551 001726 126227 000000G 000000G CMPB C.STS(R2),#CS.REM ; REMOTE ?
552 001734 001372 BNE 20$, ; NO
553 001736 026201 000010 CMP C.ADD(R2),R1 ; THIS NDDE ?
554 001742 001367 BNE 20$, ; NO
555 001744 016246 000006 MOV C.NAM+2(R2),-(SP) ; PUSH PART 2
556 001750 016201 000004 MOV C.NAM(R2),R1 ; GET PART 1
557
558 001754 112720 000040 25$: MOV #<'>,(R0)+ ; INSERT DELIMITER
559 001760 112720 000050 MOV #<'>,(R0)+ ;
560 001764 004767 000000G CALL $C5TA ; OUTPUT IT
561 001770 012601 MOV (SP)+,R1 ; RECOVER PART 2
562 001772 004767 000000G CALL $C5TA ; OUTPUT IT
563 001776 112720 000051 MOV #<'>,(R0)+ ; INSERT DELIMITER
564
565 002002 000207 40$: RETURN
566
567 ;+
568 ;** PUTR50 - OUTPUT RAD50 VALUE

```

```

172 000666 012701 000142' MOV #PARMSG,R1 ; POINT R1 AT INPUT STRING
173 000672 004767 000000G CALL $EDMSG ; EDIT THE STRING
174 000676 PRINT # $BUFF ; PRINT THE MESSAGE
175
176 000706 012702 000530' MOV #BUF,R2 ; POINT AT ARGUMENT BLOCK
177 000712 016712 000000G MOV $BYT,(R2) ; MOVE IN THE BYTE-AREA EXTENSION
178 000716 021267 000000G CMP (R2),$BYTMN ; CHECK IT AGAINST BYTE-AREA MINIMUM
179 000722 101002 BHI 7$; IF GT, OK
180 000724 016712 000000G MOV $BYTMN,(R2) ; ELSE REPORT MINIMUM VALUE
181
182 000730 012762 000007' 000002 7$: MOV #TOPLN,2(R2) ; ASSUME TOPDOWN LOADING
183 000736 012762 000266' 000004 MOV #TOP,4(R2) ; ...
184 000744 032765 000000G 000000G BIT #CS.TOP,C.STS(R5) ; TOPDOWN LOADING ?
185 000752 001006 BNE 15$; IF NE, YES... OKAY
186 000754 012762 000010' 000002 MOV #FIRLN,2(R2) ; ELSE, SAY THIS IS FIRSTFIT
187 000762 012762 000275' 000004 MOV #FIRST,4(R2) ; ...
188
189 000770 012700 000000G 15$: MOV # $BUFF,R0 ; POINT R0 AT OUTPUT BUFFER
190 000774 012701 000213' MOV #PAR2MS,R1 ; POINT R1 AT MESSAGE STRING
191 001000 004767 000000G CALL $EDMSG ; FORMAT THE MESSAGE
192 001004 PRINT # $BUFF ; PRINT THE MESSAGE
193
194 ; SEARCH FOR THE BUFFER TEMPLATE
195
196 001014 011404 20$: MOV (R4),R4 ; GET NEXT TEMPLATE
197 001016 001445 BEQ 102$; IF EQ, END OF TEMPLATES
198 001020 122764 000000G 000000G CMBP #CS.BUF,C.STS(R4) ; IS THIS A BUFFER TEMPLATE ?
199 001026 001372 BNE 20$; IF NE, NO - KEEP LOOKING
200
201 ; DISPLAY THE BUFFER CHARACTERISTICS
202
203 001030 012701 000530' 30$: MOV #BUF,R1 ; POINT R1 AT ARGUMENT BLOCK
204 001034 016421 000004 MOV C.CBNM(R4),(R1)+ ; STORE THE CCB NUMBER
205 001040 016421 000014 MOV C.SBNM(R4),(R1)+ ; STORE THE SDB NUMBER
206 001044 016421 000010 MOV C.RBNM(R4),(R1)+ ; STORE THE RDB NUMBER
207 001050 016421 000012 MOV C.RBSZ(R4),(R1)+ ; STORE THE RDB SIZE
208 001054 016421 000020 MOV C.THSH(R4),(R1)+ ; STORE THE THRESHOLD VALUE
209
210 ; FORMAT THE BUFFER MESSAGES
211
212 001060 012703 000542' MOV #MSTBL,R3 ; POINT R3 AT THE MESSAGE TABLE
213 001064 012702 000530' MOV #BUF,R2 ; POINT R2 AT THE ARGUMENT BLOCK
214 001070 012301 40$: MOV (R3)+,R1 ; POINT R1 AT THE MESSAGE STRING
215 001072 001411 BEQ 50$; IF EQ, END OF MESSAGES
216 001074 012700 000000G MOV # $BUFF,R0 ; POINT R0 AT OUTPUT BUFFER
217 001100 004767 000000G CALL $EDMSG ; FORMAT THE MESSAGE
218 001104 PRINT # $BUFF ; PRINT THE MESSAGE
219 001114 000765 BR 40$; REPEAT
220 001116 000207 50$: RETURN ; RETURN TO CALLER
221
222 ; ERROR CONDITIONS
223
224 001120 101$: ERROR$ #ERRPAR ; FAILED TO FIND PARTITION TEMPLATE
225 001130 BR 111$; AND RETURN
226 001132 102$: ERROR$ #ERRBUF ; FAILED TO FIND BUFFER TEMPLATE
227 001142 000207 111$: RETURN
228

```

265  
 266  
 267  
 268  
 269  
 270  
 271  
 272  
 273  
 274  
 275  
 276  
 277  
 278  
 279  
 280  
 281  
 282  
 283  
 284  
 285  
 286  
 287  
 288  
 289  
 290  
 291  
 292  
 293

000176 016700 177576  
 000202 016701 000000G  
 000206 010160 000006  
 000212 020167 177570  
 000216 103403  
 000220 020167 177564  
 000224 101402  
 000226 062716 000002  
 000232 000207

```

.SBTTL SETCBS - SET CCB SIZE
+
*** - SETCBS - SET CCB SIZE
REASONABLE CHECKS ARE PERFORMED ON THE CURRENT VALUE. IF THE VALUE IS
NOT IN RANGE, THE C-BIT IS SET.
INPUT:
TEMP = TEMPLATE ADDRESS
$LIMITB = ADDRESS OF LIMITS TABLE
.PNUMB = CURRENT VALUE
.PNUMH = CURRENT VALUE (HIGH ORDER)
OUTPUT:
C-BIT = SUCCESS/FAILURE
VALUE IS STORED IN TEMPLATE.
;-
SETCBS:
MOV TEMP,R0 ; POINT AT TEMPLATE
MOV .PNUMB,R1 ; GET THE SIZE OF EACH CCB
MOV R1,C.CBSZ(R0) ; STORE THE CCB SIZE
CMP R1,LOCBSZ ; IS THE VALUE IN RANGE ?
BLO 10$; IF LO, NO - SET C-BIT
CMP R1,HICBSZ ; STILL IN RANGE ?
BLOS 20$; IF LOS, YES - JUST LEAVE
10$: ADD #2,(SP) ; REJECT THE TRANSITION (SETS C-BIT)
20$: RETURN

```

.TITLE TMPCN - PARSE OUTGOING CHANNEL DEFINITION  
.IDENT /V05.00/  
.ENABL LC

... COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

... THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

... THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

... DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

... MODULE DESCRIPTION:

STATE TABLE TO PARSE OUTGOING CHANNEL DEFINITION

... DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

... IDENT HISTORY:

- 1.00 14-AUG-81  
DECNET-11M/S V3.1  
DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 ;****
65 ; MACRO CALLS
66 ;****
67
68 .MCALL ISTAT$,STAT$,TRAN$
69
70 ;****
71 ; LOCAL DATA
72 ;****
73
74 000000 TEMP: .BLKW 1 ; ADDRESS OF CONTROLLER TEMPLATE
75 ;
76 ; DEFINE CONTROLLER TEMPLATE
77 ;
78 000000 .ASECT
79 000000 .=0
80 000000 .BLKW 1 ; C.LNK
81 000002 .BLKW 1 ; C.STS
82 000004 C.CNT: .BLKW 1 ; CONTROLLER NUMBER
83 000006 C.VEC: .BLKW 1 ; VECTOR ADDRESS
84 000010 C.CSR: .BLKW 1 ; CSR ADDRESS
85 000012 C.PRI: .BLKW 1 ; CONTROLLER PRIORITY
86 000014 C.URM: .BLKW 1 ; UNIBUS RUN MASK
87 000016 C.LEN=.
88 000002 .PSECT
89

```

.TITLE TMPCUG - PARSE CLOSED USER GROUP  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE CLOSED USER GROUP DESCRIPTOR

#### DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0



```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 ;****
65 ; MACRO CALLS
66 ;****
67
68 .MCALL ISTAT$,STAT$,TRAN$
69
70 ;****
71 ; LOCAL DATA
72 ;****
73
74 FLAG: .BLKW 1 ; FLAGS WORD
75 TEMP: .BLKW 1 ; ADDRESS OF DDM TEMPLATE
76 ;
77 ; DEFINE DDM TEMPLATE
78 ;
79 .ASECT
80 .=0
81 .BLKW 1 ; C.LNK
82 .BLKW 1 ; C.STS
83 C.NAM: .BLKW 1 ; DDM PROCESS NAME
84 C.FLG: .BLKW 1 ; DDM FLAGS WORD
85 C.PRI: .BLKW 1 ; DDM PROCESS PRIORITY
86 C.NLT: .BLKB 1 ; NUMBER OF LINE TABLES IN PROCESS SPACE
87 C.NCT: .BLKB 1 ; NUMBER OF CONTROLLER TABLES IN PROCESS SPACE
88 C.LEN=.
89 .PSECT

```

|     |     |                                                         |
|-----|-----|---------------------------------------------------------|
| 4-  | 44  | MACRO DEFINITIONS                                       |
| 5-  | 63  | DEC\$DF PARSE TABLES                                    |
| 6-  | 141 | \$TMDEC - SETUP DEC TEMPLATE                            |
| 7-  | 169 | SETLKS - NUMBER OF LOGICAL LINKS PROCESS SHOULD SUPPORT |
| 8-  | 194 | SETINDC - NUMBER OF NODE COUNTERS                       |
| 9-  | 222 | SETINC - INCOMING TIMER                                 |
| 10- | 248 | SETOUT - OUTGOING TIMER                                 |
| 11- | 274 | SETINA - INACTIVITY TIMER                               |
| 12- | 300 | SETDEF - DELAY FACTOR                                   |
| 13- | 325 | SETDWF - DELAY WEIGHT                                   |
| 14- | 351 | SETIPL - INPUT PACKET LIMITER                           |
| 15- | 376 | SETLST - LINK SERVICE THRESHOLD                         |
| 16- | 401 | SETRET - RETRANSMIT FACTOR                              |
| 17- | 427 | SETSEG - ECL SEGMENT SIZE                               |

```

351 .SBTTL SETIPL - INPUT PACKET LIMITER
352 ;+
353 ;*** - SETIPL - SETUP INPUT PACKET LIMITER
354 ;
355 ;THIS ACTION ROUTINE SETS UP THE INPUT PACKET LIMITER FOR XPT
356 ;
357 ;INPUT:
358 ;.PNUMB = INPUT PACKET LIMITER
359 ;.PNUMH = (HIGH ORDER)
360 ;
361 ;OUTPUT:
362 ;THE INPUT PACKET LIMITER IS STORED IN C.IPL OF THE TEMPLATE
363 ;
364 ;-
365
366 000400 SETIPL: MOV TEMP,R0 ; GET START OF TEMPLATE
367 000400 016700 177374 000000G 000020 MOVB .PNUMB,C.IPL(R0) ; STORE INPUT PACKET LIMITER
368 000404 116760 000000G 000020 TST .PNUMH ; IS IT ONLY A WORD VALUE ?
369 000412 005767 000000G 000020 BNE 10$; IF NE, NO - REJECT
370 000416 001003 000000G 000020 TSTB .PNUMB+1 ; IS IT A BYTE VALUE ?
371 000420 105767 000001G 000020 BEQ 20$; IF EQ, YES - OKAY
372 000424 001402 000000G 000020 ADD #2,(SP) ; ELSE, REJECT TRANSITION
373 000426 062716 000000G 000020 10$:
374 000432 000207 000000G 000020 20$: RETURN

```

149  
150 000004

STATE\$

SHOLIN      CREATED BY MACRO   ON 29-JUN-85 AT 05:26      PAGE 4      J 1  
MACRO CROSS REFERENCE      CREF      04.00

MACRO NAME      REFERENCES

|         |        |       |       |       |       |       |       |       |                  |
|---------|--------|-------|-------|-------|-------|-------|-------|-------|------------------|
| CHADF\$ | #4-103 | 4-105 |       |       |       |       |       |       |                  |
| EPRINT  | #4-77  |       |       |       |       |       |       |       |                  |
| ERROR\$ | #4-66  | 6-417 | 9-625 |       |       |       |       |       |                  |
| PRINT   | #4-59  | 5-305 | 6-383 | 6-394 | 6-404 | 6-409 |       |       |                  |
| PRO     | #4-90  | 4-246 | 4-248 | 4-249 | 4-250 | 4-251 | 4-252 | 4-255 | 4-256      4-257 |

379  
380  
381  
382  
383  
384  
385  
386  
387  
388  
389  
390  
391

.SBTTL CKPREV

CKPREV - CHECK TO SEE IF PREVIOUS EVENT WAS SET AND IF IT WAS, DISPLAY IT

INPUTS:

FLAGS - FLAGS BYTE

PREVNT - VALUE OF PREVIOUS EVENT TYPE IF PREVIOUS EVENT TYPE WAS SET

OUTPUTS:

FLAGS - F.PREV IS CLEARED

THE PREVIOUS EVENT VALUE IS DISPLAYED IF NECESSARY

393  
394  
395  
396  
397  
398  
399  
400  
401

001320 132767 000002 176765  
001326 001411  
001330 116701 176760  
001334 001403  
001336 005002  
001340 004767 000000G  
001344 142767 000002 176741 10\$  
001352 000207 20\$

CKPREV: BITB #F.PREV,FLAGS ; WAS PREVIOUS EVENT TYPE SET  
BEQ 20\$ ; BR IF NO  
MOVB PREVNT,R1 ; GET EVENT TYPE NUMBER TO DISPLAY  
BEQ 10\$ ; BR IF FIRST NUMBER IN RANGE  
CLR R2 ; SUPPRESS ZEROES  
CALL \$CBDMG ; CONVERT TO ASCII  
BICB #F.PREV,FLAGS ; INDICATE THIS EVENT TYPE NOT SET  
RETURN

```

264 .SBTTL
265 .SBTTL $SHXPR - SHOW MODULE X25-PROTOCOL
266
267 *** - $SHXPR - SHOW MODULE X25-PROTOCOL
268
269 INPUT:
270 $OPRND = TYPE OF REQUEST
271
272 OUTPUT:
273 THE HEADER IS PRINTED AND THE SHO INFORMATION ROUTINE
274 IS DISPATCHED TO
275
276
277 $SHXPR::
278 002042 012705 002104' MOV #XPRDTB,R5 ; POINT TO DISPATCH TABLE
279 002046 012704 000000G MOV #$HDRBK,R4 ; POINT TO ARGUMENT BLOCK
280 002052 012724 000014 MOV #XPRL,(R4)+ ; MODULE IS X25-PROTOCOL
281 002056 012724 001646' MOV #XPR,(R4)+ ; ...
282 002062 016703 000000G MOV $OPRND,R3 ; GET REQUEST TYPE
283 002066 042703 000000C BIC #^C<XP.DTE!XP.KND!XP.CUG!XP.KNG>,R3 ; CLEAR DONT CARE BITS
284 002072 012767 177777 175712 MOV #-1,OPTION ; WANT ALL OPTIONS
285 002100 CALLR DISPATCH ; DISPATCH TO ROUTINE
286
287 ; DISPATCH TABLE
288
289
290 002104 002156' 000000 000017 XPRDTB: .NLIST BEX
291 002114 002326' 000000G 000017 .WORD XPRCHA, 0, CHARL, CHAR ; CHARACTERISTICS
292 002124 002326' 000000G 000012 .WORD XPRDTE, XP.DTE, CHARL, CHAR ; DTE
293 002134 002774' 000000G 000017 .WORD XPRDTE, XP.KND, KDTEL, KDTE ; KNOWN DTES
294 002144 002774' 000000G 000014 .WORD XPRCUG, XP.CUG, CHARL, CHAR ; GROUP
295 002154 000000 .WORD XPRCUG, XP.KNG, KCUGL, KCUG ; KNOWN GROUPS
296 .WORD 0
297 .LIST BEX

```

```

704 .SBTTL
705 .SBTTL DISPATCH - DISPATCH TO SHOW ROUTINE
706 ;+
707 ;** DISPATCH - DISPATCH TO SHOW ROUTINE
708 ;
709 ; INPUTS R5 = DISPATCH TABLE ADDRESS
710 ; R4 = ARGUMENT BLOCK ADDRESS
711 ; R3 = REQUEST CODE
712 ;
713 ; OUTPUT $HDRBK = HEADER INFORMATION
714 ; HDRPRT = 0
715 ; THE ROUTINE SHOW INFORMATION ROUTINE IS ENTERED.
716 ;
717 ; REGISTERS R2,R4,R5 ARE CORRUPTED.
718 ;
719 ; -
720
721 DISPATCH:
722 003750 005067 174024 CLR HDRPRT ; NO INFORMATION, YET
723 003754 000401 BR 20$; NO HEADER INFO TO SKIP, FIRST PASS
724 003756 022525 10$: CMP (R5)+,(R5)+ ; SKIP HEADER
725 003760 012502 20$: MOV (R5)+,R2 ; GET ROUTINE ADDRESS
726 003762 001415 BEQ 30$; IF EQ, NO SUCH REQUEST
727 003764 022503 CMP (R5)+,R3 ; COMPARE REQUEST TYPES
728 003766 001374 BNE 20$; IF NE, KEEP LOOKING
729 003770 012524 MOV (R5)+,(R4)+ ; ELSE STUFF HEADER INFORMATION
730 003772 012524 MOV (R5)+,(R4)+ ;
731 003774 004712 CALL (R2) ; DISPATCH TO SHOW ROUTINE
732 003776 005767 173776 TST HDRPRT ; ANYTHING PRINTED ?
733 004002 001004 BNE 25$; IF NE, YES
734 004004 PRINT #NOINF ; ELSE NO INFORMATION
735 004014 000207 25$: RETURN
736
737 30$: ERROR$ #ERRREQ ; UNKNOWN REQUEST TYPE
738 SEC
739 RETURN
740

```



SHOMOD CREATED BY MACRO ON 29-JUN-85 AT 05:27 PAGE 6 J 5

MACRO CROSS REFERENCE CREF 04.00

| MACRO NAME | REFERENCES                                                        |
|------------|-------------------------------------------------------------------|
| BEGTXT     | #4-103 5-187 5-198 5-209 5-216 5-222 5-236                        |
| CALLR      | #4-59 6-285 11-529 12-562 14-623                                  |
| ENDTXT     | #5-187 5-194 #5-198 5-202 #5-209 5-212 #5-216 5-218 #5-222 5-231  |
| EPRINT     | #4-83 5-236                                                       |
| ERRORS     | #4-72 18-737                                                      |
| OPTION     | #4-147 21-880 21-881 21-882 21-883 21-888 21-889                  |
| PRINT      | #4-65 9-449 9-457 18-734 19-769                                   |
| STRING     | #4-96 5-242 5-243 5-244 5-245 5-246 5-247 5-248 5-249 5-250       |
| TEXT       | 5-251 5-187 5-188 5-189 5-190 5-191 5-192 5-193 5-198 5-199 5-200 |
|            | 5-201 #5-209 5-211 5-211 #5-216 5-217 #5-222 5-223 5-224 5-225    |
|            | 5-226 5-227 5-228 5-229 5-230 #5-236 5-237 5-237                  |
| \$\$\$BLK  | #4-139 5-194 5-194 5-194 5-194 5-194 5-194 5-202 5-202 5-202      |
|            | 5-212 5-212 5-218 5-231 5-231 5-231 5-231 5-231 5-231             |
|            | 5-231 5-238                                                       |
| \$\$\$FMT  | #4-143 5-194 5-194 5-194 5-194 5-194 5-194 5-202 5-202 5-202      |
|            | 5-212 5-212 5-218 5-231 5-231 5-231 5-231 5-231 5-231             |
|            | 5-231 5-238                                                       |
| \$\$\$STR  | #4-134 5-188 5-189 5-190 5-191 5-192 5-193 5-199 5-200 5-201      |
|            | 5-210 5-211 5-217 5-223 5-224 5-225 5-226 5-227 5-228 5-229       |
|            | 5-230 5-237                                                       |

.TITLE SHOREM - SHOW REMOTE NODE CHARACTERISTICS  
.IDENT /V05.00/  
.ENABL LC

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1984, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

CFE - SHOW REMOTE DEFAULTS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

```

568
569
570
571
572
573
574
575 002004 066702 176334
576 002010 010205
577 002012 012501
578 002014 004767 000000G
579 002020 077304
580 002022 000207
581
582
583
584
585
586
587
588
589
590 002024 012701 000306'
591 002030 112120
592 002032 001376
593 002034 005300
594 002036 000207
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609 002040 066702 176300
610 002044 011202
611 002046
612 002056 012701 000334'
613 002062 010211
614 002064 000403
615
616 002066 066702 176252
617 002072 010201
618 002074 012702 060010
619 002100
620 002112 050302
621 002114 000167 000000G
622
623
624

; INPUTS: R0 = BUFFER POINTER
; R2 = OFFSET
; R3 = WORD COUNT

; OUTPUTS: R0 IS UPDATED. R4 IS PRESERVED.

PUTR50: ADD SERPT,R2 ; POINT TO SOURCE
 MOV R2,R5 ; COPY ADDRESS
10$: MOV (R5)+,R1 ; NEXT WORD
 CALL $C5TA1 ; CONVERT IT
 SOB R3,10$; CONTINUE
 RETURN ; DONE

;+
; ** PUTSNT - OUTPUT SERVICE NODE VERSION
; INPUTS: R0 = BUFFER POINTER
; R2 = SOURCE FIELD OFFSET
; OUTPUTS: R0 IS UPDATED. R4 IS PRESERVED.

PUTSNT: MOV #PHASE3,R1 ; POINT TO STRING
10$: MOV (R1)+,(R0)+ ; MOVE
 BNE 10$; STRING
 DEC R0 ; DONE
 RETURN

;+
; ** PUTOCT - OUTPUT OCTAL VALUE
; INPUTS: R0 = BUFFER POINTER
; R2 = SOURCE FIELD OFFSET
; R3 = BYTE COUNT
; OUTPUTS: R0 IS UPDATED

;--
 .ENABL LSB

PUTOC2: ADD SERPT,R2 ; POINT TO SOURCE
 MOV (R2),R2 ; GET IT
 ASL 4,R2 ; CONVERT 1K UNITS TO 64. BYTE UNITS
 MOV #SCR,R1 ; POINT TO SCRATCH BUFFER
 MOV R2,(R1) ; FILL IN SCRATCH BUFFER
 BR 10$; JOIN COMMON CODE

PUTOCT: ADD SERPT,R2 ; GET SOURCE ADDRESS
 MOV R2,R1 ; POINT TO SOURCE
10$: MOV #4000*12,+8.,R2 ; FIELD WIDTH, RADIX
 ASL 5,R3 ; SHIFT BYTE COUNT LEFT
 BIS R3,R2 ; FOLD IN BYTE COUNT
 CALLR $CBTA3 ; CONVERT BINARY TO ASCII

 .DSABL LSB

```

SHOSYS - CFE SHOW SYSTEM PARAM MACRO V05.03b Saturday 29-Jun-85 05:28 Page 3-4  
SHSYS - SHOW SYSTEM CHARACTERISTICS

229

000001

.END

295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326

000234  
000234 016700 177540  
000240 016701 000000G  
000244 010160 000010  
000250 010167 000000G  
000254 010167 177552  
000260 005367 177546  
000264 020167 177522  
000270 103403  
000272 020167 177516  
000276 101402  
000300 062716 000002  
000304 000207

```
.SBTTL SETRBN - SET RDB NUMBER
+
*** - SETRBN - SET RDB NUMBER
REASONABLE CHECKS ARE PERFORMED ON THE CURRENT VALUE. IF THE VALUE IS
NOT IN RANGE, THE C-BIT IS SET.
INPUT:
TEMP = TEMPLATE ADDRESS
$LIMITB = ADDRESS OF LIMITS TABLE
.PNUMB = CURRENT VALUE
.PNUMH = CURRENT VALUE (HIGH ORDER)
OUTPUT:
C-BIT = SUCCESS/FAILURE
VALUE IS STORED IN TEMPLATE.
-
SETRBN:
MOV TEMP,R0 ; POINT AT TEMPLATE
MOV .PNUMB,R1 ; GET THE NUMBER OF RDB'S
MOV R1,C.RBNM(R0) ; STORE THE RDB NUMBER
MOV R1,$RDBNM ; SAVE THE RDB NUMBER FOR LATER
MOV R1,H1THSH ; SET MAXIMUM NUMBER FOR RDB THRESHOLD
DEC H1THSH ; ..MINUS ONE
CMP R1,LORBNM ; IS THE VALUE IN RANGE ?
BLO 10$; IF LO, NO - SET C-BIT
CMP R1,H1RBNM ; STILL IN RANGE ?
BLOS 20$; IF LOS, YES - JUST LEAVE
ADD #2,(SP) ; REJECT THE TRANSITION (SETS C-BIT)
10$: RETURN
20$: RETURN
```

```

57 .SBTTL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61 .MACRO PRINT TEXT
62 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
63 CALL $PRINT ; PRINT MESSAGE
64 .ENDM PRINT
65
66 .MACRO ERROR$ TEXT
67
68 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
69 ; MESSAGE STRING.
70
71 .IF DIF <TEXT><R0>
72 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
73 .ENDC
74 CALL $CFERR ; PRINT ERROR MESSAGE
75 .ENDM ERROR$
76
77 .MACRO EPRINT TEXT
78
79 ; PRINT TEXT ON ERROR LUN
80
81 .IF DIF <TEXT><R0>
82 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
83 .ENDC
84 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
85 .ENDM EPRINT
86
87 ; GET NUMBER WITH RANGE CHECKING
88
89 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
90 MOV OFFS,R2
91 MOV LO,R0
92 MOV HI,R1
93 .IF NB <OPT>
94 BIS OPT,@.FLAGS
95 .ENDC
96 JMP $GTNUM
97 .ENDM
98
99 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
100 MOV OFFS,R2
101 MOV LO,R0
102 MOV HI,R1
103 .IF NB <OPT>
104 BIS OPT,@.FLAGS
105 .ENDC
106 JMP $GTNMB
107 .ENDM
108
109 ; GET RAD50 WITH FIELD SIZE CHECKING
110
111 .MACRO $GTR50 OFFS,LEN,OPT
112 MOV OFFS,R2
113

```

```

91 ;****
92 ; INITIALIZE TABLE GENERATION
93 ;****
94
95 000002 ISTAT$ CNTSTB,CNTKTB
96
97 000002 STATES$
98 000002 TRANS$ $NUMBR,,SEICNT
99
100 000002 STATES$
101 000002 TRANS$ <','>
102
103 000002 STATES$
104 000002 TRANS$ $NUMBR,,SETVEC
105
106 000002 STATES$
107 000002 TRANS$ <','>
108
109 000002 STATES$
110 000002 TRANS$ $NUMBR,,SETCSR
111
112 000002 STATES$
113 000002 TRANS$ <','>
114
115 000002 STATES$
116 000002 TRANS$ $NUMBR,,SETPRI
117
118 000002 STATES$
119 000002 TRANS$ <','>
120
121 000002 STATES$; UNIBUS RUN MASK (M+)
122 000002 TRANS$ $NUMBR,RCDEV,SETURM
123 000002 TRANS$ $LAMDA
124
125 000002 STATES$ RCDEV
126 000002 TRANS$ <','>
127
128 000002 STATES$
129 000002 TRANS$ $ALPHA
130 000002 STATES$
131 000002 TRANS$ $ALPHA
132
133 000002 STATES$
134 000002 TRANS$ $EOS,$EXIT
135
136 000002 STATES$

```

```

57 .SBTTL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61
62 .MACRO PRINT TEXT
63 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM
66
67 .MACRO ERROR$ TEXT
68
69 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
70 ; MESSAGE STRING.
71
72 .IF DIF <TEXT><R0>
73 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
74 .ENDC
75 CALL $CFERR ; PRINT ERROR MESSAGE
76 .ENDM
77
78 .MACRO EPRINT TEXT
79
80 ; PRINT TEXT ON ERROR LUN
81
82 .IF DIF <TEXT><R0>
83 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
84 .ENDC
85 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
86 .ENDM
87
88 ; GET NUMBER WITH RANGE CHECKING
89
90 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
91 MOV OFFS,R2
92 MOV LO,R0
93 MOV HI,R1
94 .IF NB <OPT>
95 BIS OPT,@.FLAGS
96 .ENDC
97 JMP $GTNUM
98 .ENDM
99
100 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
101 MOV OFFS,R2
102 MOV LO,R0
103 MOV HI,R1
104 .IF NB <OPT>
105 BIS OPT,@.FLAGS
106 .ENDC
107 JMP $GTNMB
108 .ENDM
109
110 ; GET RAD50 WITH FIELD SIZE CHECKING
111
112 .MACRO $GTR50 OFFS,LEN,OPT
113 MOV OFFS,R2

```



```

91 ;****
92 ; INITIALIZE TABLE GENERATION
93 ;****
94
95 000004 ISTAT$ DDMSTB,DDMKTB
96
97 000004 STATES$
98 000004 TRANS$ $RAD50,,SETNAM
99
100 000004 STATES$
101 000004 TRANS$ '<','>'
102
103 000004 STATES$
104 000004 TRANS$!ZF
105
106 000004 STATES$
107 000004 TRANS$ 'DDM',,,ZF.DDM,FLAG
108
109 000004 STATES$ BITS
110 000004 TRANS$ '! ,BIT0
111 000004 TRANS$ $LAMDA,PRIOR,SETFLG
112
113 000004 STATES$ BIT0
114 000004 TRANS$!ZF
115
116 000004 STATES$
117 000004 TRANS$ 'TIM',BITS,,ZF.TIM,FLAG
118 000004 TRANS$ 'MFL',BITS,,ZF.MFL,FLAG
119 000004 TRANS$ 'DLC',BITS,,ZF.DLC,FLAG
120 000004 TRANS$ 'MUX',BITS,,ZF.MUX,FLAG
121 000004 TRANS$ 'LMC',BITS,,ZF.LMC,FLAG
122 000004 TRANS$ 'KMX',BITS,,ZF.KMX,FLAG
123 000004 TRANS$ 'COU',BITS,,ZF.COU,FLAG
124 000004 TRANS$ 'PSE',BITS,,ZF.PSE,FLAG
125 000004 TRANS$ 'DIA',BITS,,ZF.DIA,FLAG
126 000004 TRANS$ 'DVP',BITS,,ZF.DVP,FLAG
127 000004 TRANS$ 'MAN',BITS,,ZF.MAN,FLAG
128
129 000004 STATES$ PRIOR
130 000004 TRANS$ '<','>'
131
132 000004 STATES$
133 000004 TRANS$ $NUMBR,,SETPRI
134
135 000004 STATES$
136 000004 TRANS$ $EOS,$EXIT
137 000004 TRANS$ '<','>'
138
139 000004 STATES$
140 000004 TRANS$ $NUMBR,NCT,SETNLT
141 000004 TRANS$ $LAMDA
142
143 000004 STATES$ NCT
144 000004 TRANS$ $EOS,$EXIT
145 000004 TRANS$ '<','>'
146
147 000004 STATES$; NUMBER OF CONTROLLER TABLES

```

.TITLE TMPDEC - CFE PARSE DEC\$DF DEFINITIONS IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE DEC\$DF DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 25-JUN-84  
MODULE CREATION
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RX V1.0

```

376 .SBTTL SETLST - LINK SERVICE THRESHOLD
377 ;+
378 ;*** - SETLST - SETUP LINK SERVICE THRESHOLD
379 ;
380 ; THIS ACTION ROUTINE SETS UP THE LINK SERVICE THRESHOLD FOR THE EXECUTOR
381 ;
382 ; INPUT:
383 ; .PNUMB = LINK SERVICE THRESHOLD
384 ; .PNUMH = (HIGH ORDER)
385 ;
386 ; OUTPUT:
387 ; THE LINK SERVICE THRESHOLD IS STORED IN C.LST OF THE iEMPLATE
388 ;
389 ; -
390
391 000434 SETLST:
392 000434 016700 177340 MOV TEMP,RO ; GET START OF TEMPLATE
393 000440 116760 000000G 000021 MOVB .PNUMB,C.LST(RO) ; STORE LINK SERVICE THRESHOLD
394 000446 005767 000000G TST .PNUMH ; IS IT ONLY A WORD VALUE ?
395 000452 001003 BNE 10$; IF NE, NO - REJECT
396 000454 105767 000001G TSTB .PNUMB+1 ; IS IT A BYTE VALUE ?
397 000460 001402 BEQ 20$; IF EQ, YES - OKAY
398 000462 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT TRANSITION
399 000466 000207 20$: RETURN

```

```

152 .SBTTL $TMDLC - SETUP DLC TEMPLATE
153
154 *** - $TMDLC - SETUP DLC TEMPLATE
155
156 INPUT:
157 R3-R5 - TPARS REGISTERS
158
159 OUTPUT:
160 C-BIT = SUCCESS/FAILURE
161 IF SUCCESS, THE DLC TEMPLATE IS STORED IN THE END OF TASK BUFFER.
162
163 -
164
165 $TMDLC::
166 000004 MOV R5, -(SP) ; SAVE R5
167 000004 010546 MOV #C.LEN, R1 ; GET LENGTH OF ALLOCATION
168 000012 004767 000014 CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
169 000016 010067 177760 MOV RC, TEMP ; STORE TEMPLATE ADDRESS
170 000022 012760 000000G 000000G MOV #CS.DLC, C.STS(R0) ; INDICATE THAT THIS IS A DLC$DF TEMPLATE
171 000030 105060 000012 CLR C.NLT(R0) ; ZERO THE LINE TABLE COUNT
172 000034 005001 CLR R1 ; IGNORE BLANKS
173 000036 012702 000000' MOV #DLCKTB, R2 ; KEYWORD TABLE
174 000042 012705 000000' MOV #DLCSTB, R5 ; STATE TABLE
175 000046 004767 000000G CALL ,TPARS ; PARSE THE REST OF THE LINE
176 000052 103002 BCC 20$; NORMAL RETURN IF NO ERROR
177 000054 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
178 000060 012605 20$: MOV (SP)+, R5 ; RESTORE R5
179 000062 000207 RETURN

```

\*\*FILE\*\*ID\*\*SHOLOG

K 1

```

SSSSSSSS HH HH 000000 LL 000000 GGGGGGGG
SSSSSSSS HH HH 000000 LL 000000 GGGGGGGG
SS HH HH 00 00 LL 00 00 GG
SS HH HH 00 00 LL 00 00 GG
SS HH HH 00 00 LL 00 00 GG
SS HH HH 00 00 LL 00 00 GG
SSSSSS HHHHHHHHHH 00 00 LL 00 00 GG
SSSSSS HHHHHHHHHH 00 00 LL 00 00 GG
SS HH HH 00 00 LL 00 00 GG GGGGGG
SS HH HH 00 00 LL 00 00 GG GGGGGG
SS HH HH 00 00 LL 00 00 GG GG
SS HH HH 00 00 LL 00 00 GG GG
SSSSSS HH HH 000000 LLLLLLLLLL 000000 GGGGGG
SSSSSS HH HH 000000 LLLLLLLLLL 000000 GGGGGG

```

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
SS SS TT
SS SS TT
SS SS TT
SS SS TT
SS SSSSS TT
SS SSSSS TT
SS SS TT
SS SS TT
SS SS TT
SS SS TT
SS SS TT
SSSSSS TT
SSSSSS TT

```

```

403 .SBTTL CHKBIT - CHECK TO SEE IF BIT IS ON
404
405 ;+
406 ; CHKBIT - CHECK TO SEE IF SPECIFIED BIT IS ON
407 ;
408 ; INPUTS:
409 ; BITNUM - BIT NUMBER TO CHECK
410 ; R5 - ADDRESS OF TEMPLATE BLOCK
411 ;
412 ; OUTPUTS:
413 ; CONDITION CODE AS DEFINED FOR BITB INSTRUCTION
414 ;
415 ; -
416
417 001354 012702 000001 CHKBIT: MOV #1,R2 ; GET A CHECK BIT IN POSITION 0
418 001360 116701 176726 MOV BITNUM,R1 ; GET BIT NUMBER
419 001364 042701 177770 BIC #177770,R1 ; MAKE IT A SHIFT COUNT
420 001370 001403 BEQ 20$; BR IF BIT POSITION 0
421 001372 006302 10$: ASL R2 ; SHIFT CHECK BIT LEFT
422 001374 005301 DEC R1 ; DECREMENT SHIFT COUNT
423 001376 003375 BGT 10$; BR IF MORE TO SHIFT
424 001400 116701 176706 MOV BITNUM,R1 ; GET BIT NUMBER
425 001404 006201 ASR R1 ; DIVIDE BY 8 TO GET BYTE INDEX
426 001406 006201 ASR R1 ; ...
427 001410 006201 ASR R1 ; ...
428 001412 060501 ADD R5,R1 ; POINT TO TEMPLATE BLOCK
429 001414 062701 000006 ADD #C.EVT,R1 ; GET ADDRESS OF EVENT MASKS
430 001420 130211 BITB R2,(R1) ; CHECK BIT
431 001422 000207 RETURN
432
433 .END
000001

```

```

299 .SBTTL XPRCHA - SHOW X25 PROTOCOL CHARACTERISTICS
300
301 ;+
302 ** XPRCHA - SHOW X25 PROTOCOL CHARACTERISTICS
303 ;
304 INPUT X3PDF, PSNDF TEMPLATES.
305 ;
306 OUTPUT THE LEVEL 3 PROTOCOL CHARACTERISTICS ARE DISPLAYED.
307 ;
308 XPRCHA:
309 002156 012705 000000G MOV #1MLST,R5 ; POINT TO TEMPLATE LIST HEAD
310 002156 011505 10$: MOV (R5),R5 ; GET NEXT TEMPLATE
311 002164 001457 BEQ 20$, ; IF EQ, NO LEVEL 3 INFO
312 002166 122765 000000G 000000G CMBP #CS.X3P,C.STS(R5) ; TEST TEMPLATE TYPE
313 002174 001372 BNE 10$, ; IF NE, WRONG ONE
314 002176 012700 000000' MOV #PRCHBK,R0 ; POINT TO ARGUMENT BLOCK
315 002202 016520 000000G MOV X3$DFP(R5),(R0)+ ; DEFAULT DATA
316 002206 016520 000000G MOV X3$MXP(R5),(R0)+ ; MAXIMUM DATA
317 002212 116520 000000G MOVVB X3$DW(R5),(R0)+ ; DEFAULT WINDOW
318 002216 105020 CLRB (R0)+ ; CLEAR EXTENDED SIGN
319 002220 116520 000000G MOVVB X3$MW(R5),(R0)+ ; MAXIMUM WINDOW
320 002224 105020 CLRB (R0)+ ; CLEAR EXTENDED SIGN
321 002226 116520 000000G MOVVB X3$RT(R5),(R0)+ ; RESET TIMER
322 002232 105020 CLRB (R0)+ ; CLEAR EXTENDED SIGN
323 002234 116520 000000G MOVVB X3$RM(R5),(R0)+ ; MAXIMUM RESETS
324 002240 105020 CLRB (R0)+ ; CLEAR EXTENDED SIGN
325 002242 116520 000000G MOVVB X3$ST(R5),(R0)+ ; RESTART TIMER
326 002246 105020 CLRB (R0)+ ; CLEAR EXTENDED SIGN
327 002250 116520 000000G MOVVB X3$SM(R5),(R0)+ ; MAXIMUM RESTARTS
328 002254 105020 CLRB (R0)+ ; CLEAR EXTENDED SIGN
329 002256 116520 000000G MOVVB X3$KT(R5),(R0)+ ; CLEAR TIMER
330 002262 105020 CLRB (R0)+ ; CLEAR EXTENDED SIGN
331 002264 116520 000000G MOVVB X3$KM(R5),(R0)+ ; MAXIMUM CLEARS
332 002270 105020 CLRB (R0)+ ; CLEAR EXTENDED SIGN
333 002272 016704 000000G MOV $PSIPT,R4 ; POINT TO PSN$DF
334 002276 016420 000000G MOV P$SNAM(R4),(R0)+ ; GET NETWORK NAME
335 002302 016420 000002G MOV P$SNAM+2(R4),(R0)+ ;
336 002306 116520 000000G MOVVB X3$CT(R5),(R0)+ ; CALL TIMER
337 002312 105020 CLRB (R0)+ ; CLEAR EXTENDED SIGN
338 002314 012703 000672' MOV #PRCHFT,R3 ; POINT TO FORMATTING TABLE
339 002320 004767 001506 CALL PRINT ; FORMAT AND PRINT THE INFORMATION
340 002324 000207 20$: RETURN
341

```

742  
 743  
 744  
 745  
 746  
 747  
 748  
 749  
 750  
 751  
 752  
 753  
 754  
 755  
 756  
 757  
 758  
 759  
 760  
 761  
 762  
 763  
 764  
 765  
 766  
 767  
 768  
 769  
 770  
 771  
 772  
 773

004032 012746 000001  
 004036 005767 173736  
 004042 001007  
 004044 005267 173730  
 004050 012701 000213'  
 004054 012702 000000G  
 004060 000407  
 004062 012301  
 004064 001416  
 004066 012302  
 004070 006316  
 004072 031667 173714  
 004076 001771  
 004100 012700 000000G  
 004104 004767 000000G  
 004110  
 004120 000760  
 004122 005726  
 004124 000207

```

 .SBTTL PRINT - PRINT FORMATTED INFORMATION
 *
 ** PRINT - PRINT FORMATTED INFORMATION
 INPUT R3 = ADDRESS OF FORMATTING TABLE
 OPTIONS = OPTIONS WORD
 OUTPUT CHARACTERISTICS PRINTED
 REGISTERS R4 AND R5 ARE PRESERVED.
 -
PRINT: MOV #1,-(SP) ; CLEAR CURRENT OPTION
 TST HDRPRT ; ALREADY PRINTED ?
 BNE 10$; IF NE, YES
 INC HDRPRT ; ELSE SET FLAG
 MOV #HDRMSG,R1 ; POINT TO FORMAT STRING
 MOV #HDRBK,R2 ; POINT TO ARGUMENT BLOCK
 BR 15$; OUTPUT HEADER
10$: MOV (R3)+,R1 ; POINT TO FORMATTING STRING
 BEQ 20$; IF EQ, NO MORE
 MOV (R3)+,R2 ; POINT TO ARGUMENT BLOCK
 ASL (SP) ; NEXT OPTION
 BIT (SP),OPTIONS ; WAS THIS OPTION SELECTED ?
 BEQ 10$; IF EQ,NO
15$: MOV #$BUFF,R0 ; POINT TO FORMATTING BUFFER
 CALL $EDMSG ; FORMAT OUTPUT
 PRINT #$BUFF ; PRINT IT
 BR 10$; UNTIL DONE
20$: TST (SP)+ ; CLEAN STACK
 RETURN

```



\*\*FILE\*\*ID\*\*SHOPRO

```

SSSSSSSS HH HH 000000 PPPPPPP RRRRRRR 000000
SSSSSSSS HH HH 000000 PPPPPPP RRRRRRR 000000
SS HH HH 00 00 PP PP RR RR 00 00
SS HH HH 00 00 PP PP RR RR 00 00
SS HH HH 00 00 PP PP RR RR 00 00
SSSSSS HHHHHHHH 00 00 PPPPPPP RRRRRRR 00 00
SSSSSS HHHHHHHH 00 00 PPPPPPP RRRRRRR 00 00
SS HH HH 00 00 PP RR RR 00 00
SS HH HH 00 00 PP RR RR 00 00
SS HH HH 00 00 PP RR RR 00 00
SSSSSS HH HH 000000 PP RR RR 000000
SSSSSS HH HH 000000 PP RR RR 000000

```

```

....
....
....
....

```

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSS TT
LL SSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLL SSSSSSS TT
LLLLLLLLL SSSSSSS TT

```

```

56 ;****
57 ; LOCAL MACROS
58 ;****
59
60 .MACRO PRINT TEXT
61 .IF DIF <TEXT><RO>
62 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
63 .ENDC
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM PRINT
66
67 .MACRO ERRORS$ TEXT
68
69 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
70 ; MESSAGE STRING.
71
72 .IF DIF <TEXT><RO>
73 MOV TEXT,RO ; GET ADDRESS OF ERROR MESSAGE
74 .ENDC
75 CALL $CFERR ; PRINT ERROR MESSAGE
76 .ENDM ERRORS$
77
78 .MACRO EPRINT TEXT
79
80 ; PRINT TEXT ON ERROR LUN
81
82 .IF DIF <TEXT><RO>
83 MOV TEXT,RO ; GET ADDRESS OF ASCIZ STRING TO PRINT
84 .ENDC
85 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
86 .ENDM EPRINT
87
88 .MACRO DLLDAT FLG=0,OFS,RTN,ARG=0,STR,EOL
89 $$$EOL = 0
90 .IF NB <'EOL'> $$$EOL = 200
91 .IF IDN <'RTN'>,<PUTFIL> $$$EOL = $$$EOL ! 100
92 .SAVE
93 .PSECT $$$STR,D,RO
94 $$$ = .
95 .ASCIZ '/'STR'/
96 .RESTORE
97 .WORD $$$; address of ASCIZ string
98 .BYTE OFS ; $$$xx offset
99 .BYTE $$$EOL!ARG ; argument for fmt routine, hi bit = EOL
100 .WORD RTN ; address of formatting routine
101 .WORD FLG ; SF.xxx flag
102 .ENDM DLLDAT
103
104 .MCALL ASR$,SAVRG,RESRG,ASL$
105
106 ;****
107 ; LOCAL DATA
108 ;****
109
110
111 ; DEFINE REMOTE TEMPLATE
112

```

```

625 ;+
626 ; ** PUTHAD- OUTPUT HARDWARE ADDRESS
627 ;
628 ; INPUT: R0 = BUFFER POINTER
629 ; R1 = SOURCE FIELD ADDRESS
630 ;
631 ; OUTPUT: R0 IS UPDATED. R4 IS PRESERVED.
632 ;
633 PUTHAD: SAVRG R0 ; SAVE REGISTER
634 ADD #18.,R0 ; POINT TO SCRATCH BUFFER
635 SAVRG R0 ; SAVE SCRATCH ADDRESS
636 CALL PUTR50 ; CONVERT IT TO ASCII IN SCRATCH BUF
637 RESRG <R1,R0> ; GET SCRATCH POINTER, BUF POINTER
638 MOV #6.,R2 ; LENGTH IN BYTES
639 10$: MOVB (R1)+,(R0)+ ; OUTPUT A
640 MOVB (R1)+,(R0)+ ; BYTE VALUE
641 MOVB #'-(R0)+ ; DELIMITER
642 SOB R2,10$; COUNT DOWN
643 DEC R0 ; SKIP OVER LAST HYPHEN
644 RETURN
645 .END
646
647 000001

```

SHOSYS - CFE SHOW SYSTEM PARAM MACRO V05.03b Saturday 29-Jun-85 05:28 <sup>K 8</sup> Page 3-5  
Symbol table

|                  |                  |                 |                   |                   |
|------------------|------------------|-----------------|-------------------|-------------------|
| BUF 000530R      | C.NAM0 000004    | ERRBUF 000040R  | 02MSG 000400R     | \$BYTMN= ***** GX |
| CS.BUF= ***** GX | C.NAM1 000010    | ERRPAR 000000R  | 03MSG 000465R     | \$BYTXT= ***** GX |
| CS.PAR= ***** GX | C.RBNM 000010    | EX\$ERR= 000002 | PARMSG 000142R    | \$CFERR= ***** GX |
| CS.TOP= ***** GX | C.RBSZ 000012    | FIRLN = 000010  | PAR2MS 000213R    | \$EDMSG= ***** GX |
| C.ALL 000016     | C.SBNM 000014    | FIRST 000275R   | POL 000524R       | \$GTBUF= ***** GX |
| C.CBNM 000004    | C.SBSZ 000016    | HEADR 000075R   | TOP 000266R       | \$PRINT= ***** GX |
| C.CBSZ 000006    | C.STS = ***** GX | MSTBL 000542R   | TOPLN = 000007    | \$SHSYS 000552RG  |
| C.EXT 000014     | C.TSHS 000020    | 01MSG 000305R   | \$BUFF = ***** GX | \$TMLST= ***** GX |
| C.LEN = 000020   |                  |                 |                   |                   |

. ABS. 000022 000 (RW,I,GBL,ABS,OVR)  
001144 001 (RW,I,LCL,REL,CON)

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 8574 Words ( 34 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:06.28

SY:SHOSYS.V2,[132,134]SHOSYS/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]SHOSYS

```

328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347 000306
348 000306 016700 177466
349 000312 016701 000000G
350 000316 010160 000012
351 000322 010167 000000G
352 000326 020167 177464
353 000332 103403
354 000334 020167 177460
355 000340 101402
356 000342 062716 000002
357 000346 000207

 .SBTTL SETRBS - SET RDB SIZE
 ;+
 *** - SETRBS - SET RDB SIZE
 ; REASONABLE CHECKS ARE PERFORMED ON THE CURRENT VALUE. IF THE VALUE IS
 ; NOT IN RANGE, THE C-BIT IS SET.
 ; INPUT:
 ; TEMP = TEMPLATE ADDRESS
 ; $LIMITB = ADDRESS OF LIMITS TABLE
 ; .PNUMB = CURRENT VALUE
 ; .PNUMH = CURRENT VALUE (HIGH ORDER)
 ; OUTPUT:
 ; C-BIT = SUCCESS/FAILURE
 ; VALUE IS STORED IN TEMPLATE.
 ; -
 SETRBS:
 MOV TEMP,RO ; POINT A1 TEMPLATE
 MOV .PNUMB,R1 ; GET THE SIZE OF EACH RDB
 MOV R1,C.RBSZ(RO) ; STORE THE RDB SIZE
 MOV R1,$RDBSZ ; SAVE THE RDB SIZE FOR LATER
 CMP R1,LORBSZ ; IS THE VALUE IN RANGE ?
 BLO 10$; IF LO, NO - SET C-BIT
 CMP R1,HIRBSZ ; STILL IN RANGE ?
 BLOS 20$; IF LOS, YES - JUST LEAVE
 ADD #2,(SP) ; REJECT THE TRANSITION (SETS C-BIT)
 10$: RETURN
 20$:

```

```
114 MOV LEN,R1
115 .IF NB <OPT>
116 BIS OPT,a.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 ;
122 ; MACRO CALLS
123 ;
124 .MCALL ISTAT$,STATE$,TRANS$,CALLR
```

138  
 139  
 140  
 141  
 142  
 143  
 144  
 145  
 146  
 147  
 148  
 149  
 150  
 151  
 152  
 153  
 154  
 155  
 156  
 157  
 158  
 159  
 160  
 161  
 162  
 163  
 164  
 165

000002  
 000002 010546  
 000004 012701 000016  
 000010 004767 000000G  
 000014 010067 177760  
 000020 005060 000014  
 000024 012760 000000G 000000G  
 000032 005001  
 000034 012702 000000'  
 000040 012705 000000'  
 000044 004767 000000G  
 000050 103002  
 000052 005267 000000G  
 000056 012605  
 000060 000207

```

.SBTTL $TMCNT - SETUP CONTROLLER TEMPLATE
*** - $TMCNT - SETUP CONTROLLER TEMPLATE
INPUT:
R3-R5 - TPARS REGISTERS
OUTPUT:
C-BIT = SUCCESS/FAILURE
IF SUCCESS, THE CONTROLLER TEMPLATE IS STORED IN THE END OF TASK BUFFER.
-
$TMCNT::
MOV R5,-(SP) ; SAVE R5
MOV #C.LEN,R1 ; GET LENGTH OF ALLOCATION
CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
MOV R0,TEMP ; STORE TEMPLATE ADDRESS
CLR C.URM(R0) ; ZERO THE UNIBUS RUN MASK
MOV #CS.CNT,C.STS(R0) ; INDICATE THAT THIS IS A CNT$DF TEMPLATE
CLR R1 ; IGNORE BLANKS
MOV #CNTKTBL,R2 ; KEYWORD TABLE
MOV #CNTSTB,R5 ; STATE TABLE
CALL .TPARS ; PARSE THE REST OF THE LINE
BCC 20$; NORMAL RETURN IF NO ERROR
INC $ERROR ; INDICATE SYNTAX ERROR
MOV (SP)+,R5 ; RESTORE R5
20$:
RETURN

```

```

114 MOV LEN,R1
115 .IF NB <OPT>
116 BIS OPT,@.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 ::
122 :: MACRO CALLS
123 ::
124 .MCALL ISTAT$,STATE$,TRAN$,CALLR
125
126

```



|            |                             |
|------------|-----------------------------|
| 148 000004 | TRANS \$NUMBR,\$EXIT,SETNCT |
| 149        |                             |
| 150 000004 | STATES ZF                   |
| 151 000004 | TRANS "ZF"                  |
| 152        |                             |
| 153 000004 | STATES                      |
| 154 000004 | TRANS ',,\$EXIT             |
| 155        |                             |
| 156 000004 | STATES                      |

```

44 .SBTTL MACRO DEFINITIONS
45
46 ;
47 ; MACRO CALLS
48 ;
49
50 .MCALL ISTAT$,STAT$,TRANS$,MSGDF$,DECTP$
51
52 000000 DECTP$ LIST ; DEFINE DEC$DF TEMPLATE OFFSETS
 ;
 ; DEFINE DEC TEMPLATE
 ;
 .ASECT
 .=0
 .BLKW 1 ; C.LNK
 .BLKW 1 ; C.STS
 C.SEG: .BLKW 1 ; SEGMENT SIZE
 C.NDC: .BLKW 1 ; MAXIMUM NODE COUNTERS
 C.LKS: .BLKW 1 ; MAXIMUM LOGICAL LINKS
 C.INCT: .BLKB 1 ; INCOMING TIMER
 C.OUTT: .BLKB 1 ; OUTGOING TIMER
 C.INAC: .BLKB 1 ; INACTIVITY TIMER
 C.RETF: .BLKB 1 ; RETRANSMIT FACTOR
 C.DELF: .BLKB 1 ; DELAY FACTOR
 C.DELW: .BLKB 1 ; DELAY WEIGHT
 C.IPL: .BLKB 1 ; INPUT PACKET LIMITER
 C.LST: .BLKB 1 ; LINK SERVICE THRESHOLD
 CD.LEN=.
 .PSECT
 MSGDF$
 ;
 ; LOCAL SYMBOLS
 ;
 ; MXNDC = 170. ; MAXIMUM NUMBER OF NODE COUNTERS
 ;
 ; LOCAL DATA
 ;
 TEMP: .BLKW 1 ; ADDRESS OF DEC TEMPLATE

```

```

401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416 000470
417 000470 016700 177304
418 000474 116760 000000G 000015
419 000502 001406
420 000504 005767 000000G
421 000510 001003
422 000512 105767 000001G
423 000516 001402
424 000520 062716 000002
425 000524 000207

 .SBTTL SETRET - RETRANSMIT FACTOR
 *** - SETRET - SETUP RETRANSMIT FACTOR
 THIS ACTION ROUTINE SETS UP THE RETRANSMIT FACTOR VALUE FOR THE EXECUTOR
 INPUT:
 .PNUMB = RETRANSMIT FACTOR
 .PNUMH = (HIGH ORDER)
 OUTPUT:
 THE RETRANSMIT FACTOR IS STORED IN C.RETF OF THE TEMPLATE
 -
SETRET:
MOV TEMP,R0 ; GET START OF TEMPLATE
MOVB .PNUMB,C.RETF(R0) ; STORE THE NUMBER OF LOGICAL LINKS
BEQ 10$; BR IF INVALID VALUE
TST .PNUMH ; IS IT ONLY A WORD VALUE ?
BNE 10$; BR IF NO, ERROR
TSTB .PNUMB+1 ; IS IT A BYTE VALUE ?
BEQ 20$; IF EQ, YES - OKAY
ADD #2,(SP) ; ELSE, REJECT TRANSITION
10$: ADD
20$: RETURN

```

```

181 .SBTTL SETNAM - SET UP PROCESS NAME
182
183 *** - SETNAM - SETUP PROCESS NAME
184
185 THIS ACTION ROUTINE SETS UP THE PROCESS NAME.
186
187 INPUT:
188 .PSTPT = START ADDRESS OF THE PROCESS NAME IN ASCII
189 .PSTCN = NUMBER OF CHARACTERS IN THE PROCESS NAME
190
191 OUTPUT:
192 C.NAM = RAD50 PROCESS NAME
193
194 -
195
196 000064 SETNAM: MOV .PSTPT,R0 ; GET ADDRESS OF ASCII PROCESS NAME
197 000064 016700 000000G MOV PC,R1 ; PERIODS ARE ACCEPTABLE
198 000070 010701 CALL $CAT5 ; CONVERT PROCESS NAME TO RAD50
199 000072 004767 000000G MOV TEMP,R0 ; GET START OF TEMPLATE
200 000076 016700 177700 MOV R1,C.NAM(R0) ; STORE PROCESS NAME IN TEMPLATE
201 000102 010160 000004 CMP #3,.PSTCN ; IS THE PROCESS NAME TOO BIG ?
202 000106 022767 000003 000000G BHS 10$; IF HIS, NO
203 000114 103002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
204 000116 062716 000002 10$: RETURN
205 000122 000207

```

SHOLOG - LIST LOGGING OPTIONS    MACRO V05.03b Saturday 29-Jun-85 <sup>L 1</sup> 05:26  
Table of contents

|     |     |                                             |
|-----|-----|---------------------------------------------|
| 4-  | 56  | MACRO DEFINITIONS                           |
| 5-  | 97  | LOCAL DATA                                  |
| 6-  | 134 | LOCAL SYMBOL DEFINITIONS                    |
| 7-  | 175 | SSHLOG - SHOW EVENT LOGGING CHARACTERISTICS |
| 8-  | 253 | PRISNK - PRINT SINK TYPE MESSAGE            |
| 9-  | 285 | PRISTA - PRINT LOGGING STATE                |
| 10- | 312 | PRTEVT - PRINT EVENT CHARACTERISTICS        |
| 11- | 379 | CKPREV                                      |
| 12- | 403 | CHKBIT - CHECK TO SEE IF BIT IS ON          |

|                  |                 |                |                  |                   |
|------------------|-----------------|----------------|------------------|-------------------|
| BITNUM 000312R   | ERRSUP 000000R  | FLAGS 000313R  | HEADR1 000112R   | SNKMSG 000166R    |
| CHKBIT 001354R   | EVTMSG 000246R  | F.ADD 000016   | MONITR 000267R   | SNKTBL 000322R    |
| CKPREV 001320R   | EX\$ERR= 000002 | F.CEV 000030   | MONL = 000007    | STAMSG 000230R    |
| COMMA = 000054   | FF.ADD= 000020  | F.CIR 000024   | MSGBLK 000316R   | STATE 000315R     |
| CONL = 000007    | FF.CIR= 000040  | F.CLS 000002   | MSKLEN= 000100   | TAB = 000011      |
| CONSLE 000302R   | FF.CON= 000001  | F.EVT 000004   | NOINFO 000207R   | TMPLOC= 000004    |
| CR = 000012      | FF.FIL= 000002  | F.FLG 000014   | OFF 000264R      | WC.LOG= ***** GX  |
| CS.EVT= ***** GX | FF.HST= 040000  | F.LEN 000016   | OFFL = 000003    | \$BUFF = ***** GX |
| CS.LOG= ***** GX | FF.LIN= 000010  | F.LIN 000020   | OM.CON= ***** GX | \$CBDMG= ***** GX |
| C.CLS 000004     | FF.MOD= 000100  | F.LNK 000000   | OM.FIL= ***** GX | \$CFERR= ***** GX |
| C.EVT 000006     | FF.MON= 000004  | F.MOD 000022   | ON 000262R       | \$EDMSG= ***** GX |
| C.LEN = 000020   | FF.MSK= 000077  | F.NINF= 000004 | ONL = 000002     | \$GTBUF= ***** GX |
| C.LN1 = 000005   | FF.PRT= 000200  | F.PREV= 000002 | PERIOD= 000056   | \$OPRND= ***** GX |
| C.SNK 000016     | FF.QUL= 000370  | F.REM 000026   | PREVNT 000314R   | \$PRINT= ***** GX |
| C.STA 000004     | FF.REM= 100000  | F.SEV 000004   | PRTEVT 001014R   | \$SHLOG 000332R   |
| C.STS = ***** GX | FILE 000276R    | F.STRT= 000001 | PRTSNK 000620R   | \$TMLST= ***** GX |
| DASH = 000055    | FILEL = 000004  | HEADR 000044R  | PRTSTA 000730R   | \$WILD = ***** GX |

. ABS. 000040 000 (RW,I,GBL,ABS,OVR)  
001424 001 (RW,I,LCL,REL,CON)

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 9256 Words ( 37 Pages)  
Size of core pool: 14440 words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:10.40  
SY:SHOLOG.V2,[132,134]SHOLOG/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]SHOLOG

```

343 .SBTTL XPRDTE - SHOW MODULE X25-PROTOCOL DTE
344
345 *** XPRDTE - SHOW MODULE X25-PROTOCOL DTE
346
347 INPUT $DTE = PARSED DTE ID STRING
348 DTE$DF TEMPLATE(S)
349
350 OUTPUT THE DTE CHARACTERISTICS ARE DISPLAYED.
351
352
353
354 XPRDTE:
355 002326 012705 000000G MOV #STMLST,R5 ; POINT TO TEMPLATE LISTHEAD
356 002332 011505 10$: MOV (R5),R5 ; GET NEXT TEMPLATE
357 002334 001510 BEQ 50$; IF EQ, END OF LIST
358 002336 122765 000000G 000000G CMBP #CS.DTE,C.STS(R5) ; TEST TEMPLATE TYPE
359 002344 001372 BNE 10$; IF NE, WRONG ONE
360 002346 032767 000000G 000000G BIT #XP.KND,$OPRND ; KNOWN DTES?
361 002354 001013 BNE 30$; IF NE, YES
362 002356 012700 000000G MOV #DTE,R0 ; ELSE POINT TO PARSED DTE STRING
363 002362 010501 MOV R5,R1 ; COPY TEMPLATE ADDRESS
364 002364 062701 000000G ADD #D$ADR,R1 ; POINT TO ID STRING IN TEMPLATE
365 002370 012702 000005 MOV #5,R2 ; LENGTH
366 002374 022021 20$: CMP (R0)+,(R1)+ ; COMPARE A WORD
367 002376 001355 BNE 10$; IF NE, NEXT TEMPLATE
368 002400 005302 DEC R2 ; COUNTDOWN
369 002402 003374 BGT 20$; UNTIL STRING COMPARED
370
371 ; DTE ADDRESS, COUNTER TIMER, MAX CIRCUITS, STATE
372
373 002404 012700 000000' 30$: MOV #DTEBK,R0 ; POINT TO ARGUMENT BLOCK
374 002410 010501 MOV R5,R1 ; COPY TEMPLATE ADDRESS
375 002412 062701 000000G ADD #D$ADR,R1 ; POINT TO ID STRING IN TEMPLATE
376 5 ;
377 MOV (R1)+,(R0)+ ; COPY DTE ID STRING
378 .ENDR
379 002430 016520 000000G MOV D$TCT(R5),(R0)+ ; COUNTER TIMER
380 002434 022020 CMP (R0)+,(R0)+ ; SKIP LINE ID FIELD FOR NOW
381 002436 012720 057760 MOV #*RON,(R0)+ ; ASSUME STATE IS ON
382 002442 105765 000000G TSTB D$STA(R5) ; CHECK STATE
383 002446 001003 BNE 35$; IF NE, STATE = ON
384 002450 012760 057266 177776 MOV #*ROFF,-2(R0) ; ELSE STATE = OFF
385 002456 016520 000000G 35$: MOV D$THSH(R5),(R0)+ ; MAXIMUM CIRCUITS
386 002462 012700 000034' MOV #LINBUF,R0 ; POINT TO OUTPUT BUFFER
387 002466 010503 MOV R5,R3 ; POINT TO
388 002470 062703 000000G ADD #D$TLIN,R3 ; LINE ID
389 002474 005067 000000G CLR $FLAGS ; CLEAR LINE OPTIONS FLAG
390 002500 032765 000000G 000000G BIT #CS.UNI,C.STS(R5) ; MUX LINE ?
391 002506 001403 BEQ 40$; IF EQ, NOPE
392 002510 052767 000000G 000000G BIS #FL.MUX,$FLAGS ; ELSE FORMAT UNIT NUMBER
393 002516 004767 001404 40$: CALL FMTLN2 ; FORMAT THE LINE ID
394 002522 162700 000034' SUB #LINBUF,R0 ; CALCULATE STRING LENGTH
395 002526 010067 000014' MOV R0,DTEBK+14 ; PUT LENGTH IN ARGUMENT BLOCK
396 002532 012767 000034' 000016' MOV #LINBUF,DTEBK+16 ; PUT STRING ADDRESS IN ARGUMENT BLOCK
397 002540 012703 001046' MOV #DTEFT,R3 ; POINT TO THE FORMATTING TABLE
398 002544 004767 001262 CALL PRINT ; OUTPUT CHARACTERISTICS
399 002550 004767 000004 CALL PRICHN ; OUTPUT CHANNEL LIST

```

```

775 .SBTTL FMTLIN - FORMAT A LINE ID
776 ;+
777 *** - FMTLIN - FORMAT A LINE ID
778 ;
779 INPUTS:
780 $FLAGS = FLAGS WORD
781 ;
782 OUTPUTS:
783 LINE ID FORMATTED INTO OUTPUT BUFFER
784 ;
785 REGISTERS:
786 R4 AND R5 ARE PRESERVED.
787 ;
788 -
789 004126 016301 000000G FMTLN2: MOV L$NAM(R3),R1 ; GET DEVICE NAME
790 004132 004767 000000G CALL $CSTA ; CONVERT NAME TO ASCII
791 004136 112720 000055 MOV #'-,(R0)+ ; INSERT DELIMITER
792 004142 116301 000000G MOV L$CNT(R3),R1 ; GET CONTROLLER NUMBER
793 004146 012702 014012 MOV #<10.!<3*4000>>,R2 ; BASE=10., WIDTH = 3
794 004152 004767 000000G CALL $CBTA ; CONVERT TO ASCII
795 004156 032767 000000G 000000G BIT #FL.MUX,$FLAGS ; IS THIS A MUX DEVICE ?
796 004164 001410 BEQ 10$; IF EQ, NO
797 004166 112720 000055 MOV #'-,(R0)+ ; INSERT DELIMITER
798 004172 116301 000000G MOV L$UNT(R3),R1 ; ELSE, GET UNIT NUMBER
799 004176 012702 014012 MOV #<10.!<3*4000>>,R2 ; BASE=10., WIDTH = 3
800 004202 004767 000000G CALL $CBTA ; CONVERT TO ASCII
801 004206 032767 000000G 10$: BIT #FL.MTP,$FLAGS ; IS THE LINE MULTI-POINT ?
802 004214 001414 BEQ 20$; IF EQ, NO
803 004216 032767 000000G 000000G BIT #FL.TRB,$FLAGS ; DID THE USER WANT TRIBUTARY ?
804 004224 001410 BEQ 20$; IF EQ, NO
805 004226 112720 000056 MOV #'-,(R0)+ ; INSERT DELIMITER
806 004232 016301 000000G MOV L$TRI(R3),R1 ; ELSE, GET TRIBUTARY NUMBER
807 004236 012702 014012 MOV #<10.!<3*4000>>,R2 ; BASE=10., WIDTH = 3
808 004242 004767 000000G CALL $CBTA ; CONVERT TO ASCII
809 004246 105010 20$: CLR (R0) ; MAKE STRING ASCII
810 004250 000207 RETURN

```



SHOPRO - LIST PROCESS CHARACTER MACRO V05.03b Saturday 29-Jun-85 05:27<sup>L 5</sup>  
Table of contents

4- 53 MACRO DEFINITIONS  
5- 111 \$SHOPRO - SHOW PROCESS CHARACTERISTICS

```

113 000000 .ASECT
114 000000 = 0
115 000000 .BLKW 1 ; C.LNK
116 000002 .BLKW 1 ; C.STS
117 000004 C.NAM: .BLKW 2 ; REMOTE NODE NAME
118 000010 C.ADD: .BLKW 1 ; REMOTE NODE ADDRESS
119 000012 C.LEN=
120 000000 .PSECT
121 ;
122 ; DEFINE SER$DF TEMPLATE
123 ;
124 000012 .ASECT
125 000000 = 0
126 000000 .BLKW 1 ; C.LNK
127 000002 .BLKW 1 ; C.FLG
128 000004 S$FLG: .BLKW 1 ; option flags
129 000006 S$ADD: .BLKW 1 ; Target node address (16 bit node address)
130 000010 S$HST: .BLKW 1 ; Host node address (16 bit node address)
131 000012 S$HAD: .BLKW 4 ; Hardware address (RAD50)
132 000022 S$CIR: .BLKW 4 ; Service circuit (extended RAD50)
133 000032 S$DEV: .BLKW 1 ; Service device (extended RAD50)
134 000034 S$PSS: .BLKW 6 ; Service password (RAD50)
135 000050 S$DPA: .BLKW 2 ; Dump address (octal binary)
136 000054 S$DPC: .BLKW 1 ; Dump count (in k bytes)
137 000056 S$LEN=
138 000000 .PSECT
139 ;
140 ; Flags definitions:
141 ;
142 000001 SF.ADD = 1 ; Target address specified
143 000002 SF.HST = 2 ; Host address specified
144 000004 SF.HAD = 4 ; Hardware address specified
145 000010 SF.CIR = 10 ; Service circuit specified
146 000020 SF.DEV = 20 ; Service device specified
147 000040 SF.PSS = 40 ; Service password specified
148 000100 SF.DPA = 100 ; Dump address specified
149 000200 SF.DPC = 200 ; Dump count specified
150 000400 SF.PH3 = 400 ; Service node type is Phase 3 IF SET
151 ;
152 ; Define FIL$DF template
153 ;
154 ;
155 000056 .ASECT
156 000000 = 0
157 000000 .BLKW 1 ; Link
158 000002 .BLKW 1 ; Flags
159 000004 F$SEC: .BLKW 1 ; Secondary loader file name
160 000006 F$TER: .BLKW 1 ; Tertiary loader file name
161 000010 F$LD: .BLKW 1 ; Load file name
162 000012 F$DIA: .BLKW 1 ; Diagnostics file name
163 000014 F$DUM: .BLKW 1 ; Dump file name
164 000016 F$LEN=
165 000000 .PSECT
166 ;
167 ; Define file descriptor offsets
168 ;
169 ;

```

|                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                                                                                                                                                                                                                                                                                                                                                                      |                                                                                                                                                                                                                                                                                                                                                                                          |                                                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Symbol table</p> <p> \$ARG 000003<br/> \$FLG 000006<br/> \$LEN = 000010<br/> \$OFS 000002<br/> \$RTN 000004<br/> \$STR 000000<br/> \$COL = 000031<br/> CS.FIL= ***** GX<br/> CS.REM= ***** GX<br/> CS.SER= ***** GX<br/> C.ADD 000010<br/> C.LEN = 000012<br/> C.NAM 000004<br/> C.STS = ***** GX<br/> DLLTBL 000350R<br/> ERRNOD= ***** GX<br/> EXENM 000265R<br/> EXENOD 001002R<br/> FILPT 000346R<br/> FORMAT 000217R </p> | <p> F\$DIA 000012<br/> F\$DUM 000014<br/> F\$LD 000010<br/> F\$LEN = 000016<br/> F\$SEC 000004<br/> F\$TER 000006<br/> F.CNT 000002<br/> F.DEV 000005<br/> F.LEN = 000043<br/> F.LNK 000000<br/> GETDLL 001246R<br/> HEADR1 000000R<br/> HEADR2 000075R<br/> HEADR3 000142R<br/> HFLAG 000332R<br/> HTBL 001102R<br/> IN\$CHA= ***** GX<br/> MS1BLK 000334R<br/> MXFSP = 000036 </p> | <p> NAMCOL= 000017<br/> NODNM 000261R<br/> PHASE3 000306R<br/> PHASE4 000320R<br/> PRIHDR 001112R<br/> PUTADR 001216R<br/> PUTFIL 001560R<br/> PUTHAD 002120R<br/> PUTNOD 001614R<br/> PUTOCT 002066R<br/> PUTOC2 002040R<br/> PUTR50 002004R<br/> PUTSNT 002024R<br/> P3NOD 000246R<br/> P4NOD 000252R<br/> SCR 000334R<br/> SERPT 000344R<br/> SF.ADD= 000001<br/> SF.CIR= 000010 </p> | <p> SF.DEV= 000020<br/> SF.DPA= 000100<br/> SF.DPC= 000200<br/> SF.HAD= 000004<br/> SF.HST= 000002<br/> SF.PH3= 000400<br/> SF.PSS= 000040<br/> SSADD 000006<br/> SSCIR 000022<br/> SSDEV 000032<br/> SSDPA 000050<br/> SSDPC 000054<br/> SSFLG 000004<br/> SSHAD 000012<br/> SSHST 000010<br/> S\$LEN = 000056<br/> SSPSS 000034<br/> WC.NOD= ***** GX<br/> \$BUFF = ***** GX </p> | <p> \$CBDMG= ***** GX<br/> \$CBTA3= ***** GX<br/> \$CFERR= ***** GX<br/> \$C5TA = ***** GX<br/> \$C5TA1= ***** GX<br/> \$EDMSG= ***** GX<br/> \$GTBUF= ***** GX<br/> \$INFO = ***** GX<br/> \$NODAD= ***** GX<br/> \$NTADD= ***** GX<br/> \$NTNAM= ***** GX<br/> \$PRINT= ***** GX<br/> \$SAVAL= ***** GX<br/> \$SHREM 000522RG<br/> \$TMLST= ***** GX<br/> \$WILD = ***** GX<br/> \$\$\$ = 000341R<br/> \$\$\$EOL= 000000<br/> \$CONER= ***** GX </p> |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

002

. ABS. 000056 000 (RW,I,GBL,ABS,OVR)  
002162 001 (RW,I,LCL,REL,CON)  
\$\$\$STR 000357 002 (RO,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 8995 Words ( 36 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:15.08  
SY:SHOREM.V2,[132,134]SHOREM/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]SHOREM

SHOSYS CREATED BY MACRO ON 29-JUN-85 AT 05:28 PAGE 1 L 8  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL  | VALUE      | REFERENCES                                       |
|---------|------------|--------------------------------------------------|
| BUF     | 000530 R   | #3-130 3-176 3-203 3-213                         |
| CS.BUF  | = ***** GX | #3-198                                           |
| CS.PAR  | = ***** GX | #3-162                                           |
| CS.TOP  | = ***** GX | #3-184                                           |
| C.ALL   | 000016     | #3-95                                            |
| C.CBNM  | 000004     | #3-103 3-204                                     |
| C.CBSZ  | 000006     | #3-104                                           |
| C.EXT   | 000014     | #3-94                                            |
| C.LEN   | = 000020   | #3-96                                            |
| C.NAMO  | 000004     | #3-92 3-165                                      |
| C.NAM1  | 000010     | #3-93 3-169 3-170                                |
| C.RBNM  | 000010     | #3-105 3-206                                     |
| C.RBSZ  | 000012     | #3-106 3-207                                     |
| C.SBNM  | 000014     | #3-107 3-205                                     |
| C.SBSZ  | 000016     | #3-108                                           |
| C.STS   | = ***** GX | #3-162 3-184 3-198                               |
| C.THSH  | 000020     | #3-109 3-208                                     |
| ERRBUF  | 000040 R   | #3-116 3-226                                     |
| ERRPAR  | 000000 R   | #3-115 3-224                                     |
| EXERR   | = 000002   | #3-114 3-115 3-116                               |
| FIRLN   | = 000010   | #3-123 3-186 3-187                               |
| FIRST   | 000275 R   | #3-122 3-123                                     |
| HEADR   | 000075 R   | #3-117 3-150                                     |
| MSTBL   | 000542 R   | #3-131 3-212                                     |
| O1MSG   | 000305 R   | #3-124 3-131                                     |
| O2MSG   | 000400 R   | #3-125 3-132                                     |
| O3MSG   | 000465 R   | #3-126 3-133                                     |
| PARMSG  | 000142 R   | #3-118 3-172                                     |
| PAR2MS  | 000213 R   | #3-119 3-190                                     |
| POL     | 000524 R   | #3-129 3-168 *3-169 *3-170                       |
| TOP     | 000266 R   | #3-120 3-183                                     |
| TOPLN   | = 000007   | #3-121 3-182                                     |
| \$BUFF  | = ***** GX | #3-149 3-153 3-171 3-174 3-189 3-192 3-216 3-218 |
| \$BYTMN | = ***** GX | #3-178 3-180                                     |
| \$BYTXT | = ***** GX | #3-177                                           |
| \$CFERR | = ***** GX | #3-224 3-226                                     |
| \$EDMSG | = ***** GX | #3-152 3-173 3-191 3-217                         |
| \$GTBUF | = ***** GX | #3-151 3-174 3-192 3-218                         |
| \$PRINT | = ***** GX | #3-153                                           |
| \$SHSYS | 000552 RG  | #3-148                                           |
| \$TMLST | = ***** GX | #3-154                                           |

```

359 .SBTTL SETSBN - SET SDB NUMBER
360 :+
361 *** - SETSBN - SET SDB NUMBER
362 :
363 REASONABLE CHECKS ARE PERFORMED ON THE CURRENT VALUE. IF THE VALUE IS
364 NOT IN RANGE, THE C-BIT IS SET.
365 :
366 INPUT:
367 TEMP = TEMPLATE ADDRESS
368 $LIMTB = ADDRESS OF LIMITS TABLE
369 .PNUMB = CURRENT VALUE
370 .PNUMB = CURRENT VALUE (HIGH ORDER)
371 :
372 OUTPUT:
373 C-BIT = SUCCESS/FAILURE
374 VALUF IS STORED IN TEMPLATE.
375 :
376 -
377
378 000350 SETSBN:
379 000350 016700 177424 MOV TEMP,R0 ; POINT AT TEMPLATE
380 000354 016701 000000G MOV .PNUMB,R1 ; GET THE SDB NUMBER
381 000360 010160 000014 MOV R1,C.SBNM(R0) ; STORE THE SDB NUMBER
382 000364 010167 000000G MOV R1,$SDBNM ; SAVE THE NUMBER OF SDB'S FOR LATER
383 000370 020167 177426 CMP R1,LOSBNM ; IS THE VALUE IN RANGE?
384 000374 103002 BHS 10$; IF HIS, YES - ELSE SET C-BIT
385 000376 062716 000002 ADD #2,(SP) ; REJECT THE TRANSITION (SETS C-BIT)
386 000402 000207 10$: RETURN

```

```
126 .SBTTL LOCAL DATA
127
128 : SYNTAX ERROR DETAIL STRINGS
129 :
130 .NLIST BEX
131 EX$SEV = 4
132 000000 000004 103 106 ERRCHN: .ASCIZ <EX$SEV>/CFE -- Error in channel number/
133 .EVEN
134 .LIST BEX
```

```

167 .SBTTL SETCNT - SET UP CONTROLLER NUMBER
168 ;+
169 ;*** - SETCNT - SETUP CONTROLLER NUMBER
170 ;
171 ;THIS ACTION ROUTINE SETS UP THE CONTROLLER NUMBER
172 ;
173 ;INPUT:
174 ;.PNUMB = CONTROLLER NUMBER
175 ;.PNUMH = (HIGH ORDER)
176 ;
177 ;OUTPUT:
178 ;C.CNT = CONTROLLER NUMBER
179 ;
180 ;-
181
182 000062 SETCNT:
183 000062 MOV TEMP,RO ; GET TEMPLATE ADDRESS
184 000066 MOV .PNUMB,C.CNT(RO) ; STORE THE CONTROLLER NUMBER IN TEMPLATE
185 000074 TST .PNUMH ; IS IT A WORD VALUE ?
186 000100 BNE 10$; IF NE, NO - REJECT
187 000102 TSTB .PNUMB+1 ; IS THE CONTROLLER NUMBER IN RANGE ?
188 000106 BEQ 20$; IF EQ, YES - OKAY
189 000110 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
190 000114 ORG 000207
10$:
20$: RETURN

```

```

128 LOCAL DATA
129 .SBTTL LOCAL DATA
130 ;
131 ; DEFINE THE CUG PARAMETER BLOCK
132 ;
133 .ASECT
134 .=0
135 .BLKW 1 ; C.LNK
136 .BLKW 1 ; C.STS
137 C.NAM: .BLKW 2 ; CUG NAME
138 C.CUG: .BLKW 1 ; GROUP NUMBER
139 C.FLG: .BLKB 1 ; FLAGS BYTE
140 C.LEN=.
141 .PSECT
142 .NLIST BEX
143 ;
144 ; SYNTAX ERROR DETAIL STRINGS
145 ;
146 .NLIST BEX
147 EX$SEV = 4
148 000000 004 103 106 ERRNAM: .ASCIZ <EX$SEV>/CFE -- Illegal CUG name/
149 000031 004 103 106 ERRUG: .ASCIZ <EX$SEV>/CFE -- Illegal group number/
150 000066 004 103 106 ERRflg: .ASCIZ <EX$SEV>/CFE -- Error in flags/
151 .EVEN
152 .LIST BEX

```



```

158 .SBTTL $TMDDM - SETUP DDM TEMPLATE
159 :+
160 *** - $TMDDM - SETUP DDM TEMPLATE
161 :
162 INPUT:
163 R3-R5 - TPARS REGISTERS
164 :
165 OUTPUT:
166 C-BIT = SUCCESS/FAILURE
167 IF SUCCESS, THE DDM TEMPLATE IS STORED IN THE END OF TASK BUFFER.
168 :
169 :-
170
171 000004 $TMDDM::
172 000004 010546 MOV R5,-(SP) ; SAVE R5
173 000006 012701 000014 MOV #C.LEN,R1 ; GET LENGTH OF ALLOCATION
174 000012 004767 000000G CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
175 000016 010067 177760 MOV R0,TEMP ; STORE TEMPLATE ADDRESS
176 000022 012760 000000G 000000G MOV #CS.DDM,C.STS(R0) ; INDICATE THAT THIS IS A DDM$DF TEMPLATE
177 000030 105060 000012 CLRB C.NLT(R0) ; ZERO THE LINE TABLE COUNT
178 000034 005001 CLR R1 ; IGNORE BLANKS
179 000036 012702 000000' MOV #DDMKTBL,R2 ; KEYWORD TABLE
180 000042 012705 000000' MOV #DDMSTBL,R5 ; STATE TABLE
181 000046 004767 000000G CALL ,TPARS ; PARSE THE REST OF THE LINE
182 000052 103002 BCC 20$; NORMAL RETURN IF NO ERROR
183 000054 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
184 000060 012605 MOV (SP)+,R5 ; RESTORE R5
185 000062 000207 RETURN

```

```

63
64
65
66 ;****
67 ; INITIALIZE TABLE GENERATION
68 ;****
69 000002 ISTAT$ DECSTB,DECKTB
70
71 000002 STATES$
72 000002 TRANS$ $NUMBR,,SETLKS ; MAXIMUM LOGICAL LINKS
73
74 000002 STATES$
75 000002 TRANS$ <','>
76
77 000002 STATES$
78 000002 TRANS$ $NUMBR,,SETNDC ; MAXIMUM NODE COUNTERS
79
80 000002 STATES$
81 000002 TRANS$ <','>
82
83 000002 STATES$
84 000002 TRANS$ $NUMBR,,SETINC ; INCOMING TIMER
85
86 000002 STATES$
87 000002 TRANS$ <','>
88
89 000002 STATES$
90 000002 TRANS$ $NUMBR,,SETOUT ; OUTGOING TIMER
91
92 000002 STATES$
93 000002 TRANS$ <','>
94
95 000002 STATES$
96 000002 TRANS$ $NUMBR,,SETINA ; INACTIVITY TIMER
97
98 000002 STATES$
99 000002 TRANS$ <','>
100
101 000002 STATES$
102 000002 TRANS$ $NUMBR,,SETDEF ; DELAY FACTOR
103
104 000002 STATES$
105 000002 TRANS$ <','>
106
107 000002 STATES$
108 000002 TRANS$ $NUMBR,,SETDEW ; DELAY WEIGHT
109
110 000002 STATES$
111 000002 TRANS$ <','>
112
113 000002 STATES$
114 000002 TRANS$ $NUMBR,,SETIPL ; INPUT PACKET LIMITER
115
116 000002 STATES$
117 000002 TRANS$ <','>
118
119 000002 STATES$

```

```

427 .SBTTL SETSEG - ECL SEGMENT SIZE
428
429 *** - SETNDC - SETUP ECL SEGMENT SIZE
430
431 THIS ACTION ROUTINE SETS UP THE ECL SEGMENT SIZE
432
433 INPUT:
434 .PNUMB = ECL SEGMENT SIZE
435 .PNUMH = (HIGH ORDER)
436
437 OUTPUT:
438 C.SEG = SEGMENT SIZE
439
440 -
441
442 SETSEG: MOV TEMP,R0 ; GET START OF TEMPLATE
443 MOV .PNUMB,C.SEG(R0) ; STORE THE SEGMENT SIZE
444 BEQ 5$; BR IF ZERO
445 ADD #N$SOVR,C.SEG(R0) ; ELSE BIAS IT
446 TST .PNUMH ; IS IT ONLY A WORD VALUE ?
447 BNE 10$; IF NE, NO - REJECT
448 BR 20$; IF EQ, YES - OKAY
449 ADD #2,(SP) ; ELSE, REJECT TRANSITION
450 RETURN
451
452 .END

```

|     |        |        |         |        |
|-----|--------|--------|---------|--------|
| 441 | 000526 |        |         |        |
| 442 | 000526 | 016700 | 177246  |        |
| 443 | 000532 | 016760 | 000000G | 000004 |
| 444 | 000540 | 001403 |         |        |
| 445 | 000542 | 062760 | 000022  | 000004 |
| 446 | 000550 | 005767 | 000000G |        |
| 447 | 000554 | 001001 |         |        |
| 448 | 000556 | 000402 |         |        |
| 449 | 000560 | 062716 | 000002  |        |
| 450 | 000564 | 000207 |         |        |
| 451 |        |        |         |        |
| 452 |        | 000001 |         |        |

SETFLG - SETUP THE FLAGS WORD

```

207 ,SBTTL SETFLG - SETUP THE FLAGS WORD
208 ;
209 ;*** - SETFLG - SETUP THE FLAGS WORD
210 ;
211 ; INPUT:
212 ; FLAG = THE FLAGS WORD
213 ; TEMP = START ADDRESS OF THE TEMPLATE
214 ;
215 ; OUTPUT:
216 ; THE FLAGS WORD IS MOVED INTO THE TEMPLATE
217 ;
218 ; -
219 ;
220 000124 SETFLG:
221 000124 016700 177652 MOV TEMP,R0 ; GET START OF TEMPLATE
222 000130 016760 177644 000006 MOV FLAG,C.FLG(R0) ; STORE THE FLAGS WORD
223 000136 000207 RETURN

```

SETPRI - SET THE PRIORITY

.TITLE SHOLOG - LIST LOGGING OPTIONS  
 .IDENT /V05.0C/  
 .NLIST BEX

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

# MODULE DESCRIPTION:

CFE - SHOW EVENT DEFAULTS

## DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

### IDENT HISTORY:

- 1.00 14-DEC-79  
 DECNET-11M/S V3.0  
 DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/s V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0

SHOLOG      CREATED BY    MACRO    ON 29-JUN-85 AT 05:26      PAGE 1      M 2

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE      | REFERENCES |         |         |         |        |         |         |         |         |
|---------|------------|------------|---------|---------|---------|--------|---------|---------|---------|---------|
| BITNUM  | 000312 R   | #5-121     | *10-325 | 10-340  | 10-346  | 10-357 | 10-367  | *10-371 | 12-418  | 12-424  |
| CHKBIT  | 001354 R   | 10-342     | #12-417 |         |         |        |         |         |         |         |
| CKPREV  | 001320 R   | 10-370     | 10-374  | #11-394 |         |        |         |         |         |         |
| COMMA   | = 000054   | #6-152     | 10-356  |         |         |        |         |         |         |         |
| CONL    | = 000007   | #5-119     | 8-269   |         |         |        |         |         |         |         |
| CONSLE  | 000302 R   | #5-118     | 5-119   | 8-270   |         |        |         |         |         |         |
| CR      | = 000012   | #6-149     |         |         |         |        |         |         |         |         |
| CS.EVT  | = ***** GX | 7-235      |         |         |         |        |         |         |         |         |
| CS.LOG  | = ***** GX | 7-193      |         |         |         |        |         |         |         |         |
| C.CLS   | 000004     | #6-162     | 10-333  |         |         |        |         |         |         |         |
| C.EVT   | 000006     | #6-163     | 12-429  |         |         |        |         |         |         |         |
| C.LEN   | = 000020   | #6-165     |         |         |         |        |         |         |         |         |
| C.LN1   | = 000005   | #6-171     |         |         |         |        |         |         |         |         |
| C.SNK   | 000016     | #6-164     | 7-237   |         |         |        |         |         |         |         |
| C.STA   | 000004     | #6-170     | 7-195   |         |         |        |         |         |         |         |
| C.STS   | = ***** GX | 7-193      | 7-235   |         |         |        |         |         |         |         |
| DASH    | = 000055   | #6-151     | 10-364  | 10-366  |         |        |         |         |         |         |
| ERRSUP  | 000000 R   | #5-103     | 7-191   |         |         |        |         |         |         |         |
| EVTMSG  | 000246 R   | #5-109     | 10-329  |         |         |        |         |         |         |         |
| EX\$ERR | = 000002   | #5-102     | 5-103   |         |         |        |         |         |         |         |
| FF.ADD  | = 000020   | #4-64      | 4-64    |         |         |        |         |         |         |         |
| FF.CIR  | = 000040   | #4-64      | 4-64    |         |         |        |         |         |         |         |
| FF.CON  | = 000001   | #4-64      | 5-129   | 7-216   | 8-267   |        |         |         |         |         |
| FF.FIL  | = 000002   | #4-64      | 5-130   | 7-219   | 8-272   |        |         |         |         |         |
| FF.HST  | = 040000   | #4-64      |         |         |         |        |         |         |         |         |
| FF.LIN  | = 000010   | #4-64      | 4-64    |         |         |        |         |         |         |         |
| FF.MOD  | = 000100   | #4-64      | 4-64    |         |         |        |         |         |         |         |
| FF.MON  | = 000004   | #4-64      | 5-131   | 7-222   |         |        |         |         |         |         |
| FF.MSK  | = 000077   | #4-64      |         |         |         |        |         |         |         |         |
| FF.PRT  | = 000200   | #4-64      | 4-64    |         |         |        |         |         |         |         |
| FF.QUL  | = 000370   | #4-64      |         |         |         |        |         |         |         |         |
| FF.REM  | = 100000   | #4-64      |         |         |         |        |         |         |         |         |
| FILE    | 000276 R   | #5-116     | 5-117   | 8-275   |         |        |         |         |         |         |
| FILEL   | = 000004   | #5-117     | 8-274   |         |         |        |         |         |         |         |
| FLAGS   | 000313 R   | #5-122     | *7-228  | 7-244   | *10-326 | 10-327 | *10-330 | 10-344  | *10-349 | *10-350 |
|         |            | 10-354     | *10-361 | 11-394  | *11-400 |        |         |         |         |         |
| F.ADD   | 000016     | #4-64      |         |         |         |        |         |         |         |         |
| F.CEV   | 000030     | #4-64      |         |         |         |        |         |         |         |         |
| F.CIR   | 000024     | #4-64      |         |         |         |        |         |         |         |         |
| F.CLS   | 000002     | #4-64      |         |         |         |        |         |         |         |         |
| F.EVT   | 000004     | #4-64      |         |         |         |        |         |         |         |         |
| F.FLG   | 000014     | #4-64      |         |         |         |        |         |         |         |         |
| F.LEN   | 000016     | #4-64      |         |         |         |        |         |         |         |         |
| F.LIN   | 000020     | #4-64      |         |         |         |        |         |         |         |         |
| F.LNK   | 000000     | #4-64      |         |         |         |        |         |         |         |         |
| F.MOD   | 000022     | #4-64      |         |         |         |        |         |         |         |         |
| F.NINF  | = 000004   | #6-145     | 7-228   | 7-244   | 10-327  | 10-330 |         |         |         |         |
| F.PREV  | = 000002   | #6-144     | 10-326  | 10-350  | 10-354  | 10-361 | 11-394  | 11-400  |         |         |
| F.REM   | 000026     | #4-64      |         |         |         |        |         |         |         |         |
| F.SEV   | 000004     | #4-64      |         |         |         |        |         |         |         |         |
| F.STRT  | = 000001   | #6-143     | 10-326  | 10-344  | 10-349  |        |         |         |         |         |
| HEADR   | 000044 R   | #5-104     | 7-199   |         |         |        |         |         |         |         |

SHOMOD - LIST MODULE CHARACTERI MACRO V05.03b Saturday 29-Jun-85 <sup>N 3</sup> 05:26 Page 8-1  
XPRDTE - SHOW MODULE X25-PROTOCOL DTE

400 002554 000666  
401 002556 000207

50\$: BR 10\$  
RETURN

SHOMOD - LIST MODULE CHARACTERI MACRO V05.03b Saturday 29-Jun-85 <sup>N 3</sup> 05:26 Page 9  
PRITCHN - PRINT DTE CHANNEL LIST

```

812 .SBTTL SHODST - SHOW DESTINATION LIST
813
814 ** SHODST - SHOW DESTINATION LIST
815
816 INPUT DTYPE = RADRO X25 OR X29
817 $OPRND = XS.KND SET IF KNOWN DESTINATIONS
818
819 OUTPUT THE DESTINATION CHARACTERISTICS ARE PRINTED.
820
821 SHODST:
822 004252
823
824 LOCATE THE DESTINATION TEMPLATE
825
826 004252 012705 000000G 10$: MOV #STMLST,R5 ; POINT TO START OF TEMPLATE LIST
827 004256 011505 MOV (R5),R5 ; NEXT
828 004260 001002 BNE 20$; IF NE, CONTINUE
829 004262 000167 000342 JMP 100$; ELSE RETURN
830 004266 122765 000000G 000000G 20$: CMPB #CS.DST,C.STS(R5) ; LOOKING FOR A DESTINATION DESCRIPTOR
831 004274 001370 BNE 10$; IF NE, WRONG TYPE OF TEMPLATE
832 004276 026567 000000G 173504 CMP D$STYP(R5),DTYPE ; CHECK DESTINATION TYPE
833 004304 001364 BNE 10$; IF NE, WRONG TYPE
834 004306 032767 000000G 000000G BIT #XS.KND,$OPRND ; KNOWN DESTINATIONS ?
835 004314 001010 BNE 30$; IF NE, YES
836 004316 026567 000000G 000000G CMP D$SNAM(R5),$DST ; COMPARE FIRST WORD OF NAME
837 004324 001354 BNE 10$; BR IF OTHER DESTINATION
838 004326 026567 000002G 000002G CMP D$SNAM+2(R5),$DST+2 ; COMPARE LAST WORD OF NAME
839 004334 001350 BNE 10$; BR IF OTHER DESTINATION
840
841 CHECK FOR SPECIAL X29 DESTINATIONS
842
843 004336 012700 002020' 30$: MOV #SPCDST-4,R0 ; POINT TO SPECIAL DESTINATION TABLE
844 004342 005720 40$: TST (R0)+ ; POINT TO NEXT SPECIAL DESTINATION
845 004344 005720 50$: TST (R0)+ ;
846 004346 001412 BEQ 60$; IF EQ, NO MORE TO CHECK
847 004350 026720 173434 CMP DTYPE,(R0)+ ; ELSE COMPARE DESTINATION TYPES
848 004354 001372 BNE 40$; IF NE, WRONG DESTINATION TYPE
849 004356 022065 000000G CMP (R0)+,D$SNAM(R5) ; COMPARE FIRST PART
850 004362 001370 BNE 50$; IF NE, NO MATCH - CONTINUE
851 004364 021065 000002G CMP (R0),D$SNAM+2(R5) ; COMPARE SECOND PART
852 004370 001365 BNE 50$; IF NE, NO MATCH - CONTINUE
853 004372 000731 BR 10$; ELSE IGNORE THIS DESTINATION IN LIST
854
855 DESTINATION ID, LIST INSERTION PRIORITY AND TASK NAME OR NUMBER
856
857 004374 012767 177007 173410 60$: MOV #*C<D$.OPT>,OPTION ; NO OPTIONS, YET
858 004402 012703 000000' MOV #DSTBK,R5 ; POINT TO ARGUMENT BLOCK
859 004406 016523 000000G MOV D$SNAM(R5),(R3)+ ; NAME
860 004412 016523 000002G MOV D$SNAM+2(R5),(R3)+ ;
861 004416 116523 000000G MOV D$SPRI(R5),(R3)+ ; LIST INSERTION PRIORITY
862 004422 105023 CLRB (R3)+ ; NIX EXTENDED SIGN
863 004424 012700 000034' MOV #CHNBUF,R0 ; POINT TO SCRATCH BUFFER
864 004430 012702 000072' MOV #CHNBUF+30,,R2 ; POINT TO SCRATCH ARGUMENT BLOCK
865 004434 012701 000275' MOV #FMTSR1,R1 ; ASSUME NUMERIC
866 004440 005012 CLR (R2) ; ZERO ARG WORD
867 004442 156512 000000G BISB D$OBJ(R5),(R2) ; SET LOW ORDER BITS OF TASK NUMBER
868 004446 001005 BNE 70$; IF NE, FORMAT IS NUMERIC

```



1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51

.TITLE SHOPRO - LIST PROCESS CHARACTERISTICS  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

CFE - SHOW PROCESS CHARACTERISTICS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-DEC-77  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/Rsx V1.0

```

170 000036 MXFSP = 30. ; Maximum Filespec length
171
172 000016 .ASECT
173 000000 . = 0
174 000000 F.LNK: .BLKW 1 ; Link word
175 000002 F.CNT: .BLKW 1 ; Use count
176 000004 .BLKB 1 ; Filespec string length
177 000005 F.DEV: .BLKB MXFSP ; Filespec string (ASCII)
178 000043 F.LEN = .
179 000000 .PSECT
180
181
182 .NLIST BEX
183 .ENABL LC
184 000000 012 116 157 HEADR: .ASCIZ <12>/Node summary as of %Y %3Z/
185 000033 012 113 156 HEADR1: .ASCIZ <12>/Known nodes summary as of %Y %3Z/
186 000075 012 116 157 HEADR2: .ASCIZ <12>/Node characteristics as of %Y %3Z/<15><12>
187 000142 012 113 156 HEADR3: .ASCIZ <12>/Known nodes characteristics as of %Y %3Z/<15><12>
188 000217 012 040 101 FORMAT: .ASCIZ <12>/ Address Name/<12>
189 000017 000031 NAMCOL = 15. ; COLUMN FOR NAME STRING
190 000031 040 045 115 P3NOD: .ASCIZ / %M/ ; COLUMN FOR CHARACTERISTICS
191 000246 040 045 115 P4NOD: .ASCIZ / %M.%M/
192 000252 040 045 115 P4NOD: .ASCIZ / %M.%M/
193 000261 045 062 122 NODNM: .ASCIZ /%2R/
194 000265 045 062 122 EXENM: .ASCIZ /%2R (Executor)/
195 000306 120 150 141 PHASE3: .ASCIZ /Phase III/
196 000320 120 150 141 PHASE4: .ASCIZ /Phase IV/
197 .EVEN
198
199 000332 HFLAG: .BLKW 1 ; HEADER PRINT FLAG
200 000334 SCR: ; SCRATCH BUFFER
201 000334 MSIBLK: .BLKW 4 ; MESSAGE ARGUMENT BLOCK
202 000344 SERPT: .BLKW 1 ; POINTER TO SER$DF
203 000346 FILPT: .BLKW 1 ; POINTER TO FIL$DF
204
205 ; DOWNLINE LOAD CONTROL TABLE
206
207 ;
208 000350 DLLTBL: DLLDAT SF.HST, $SHST, PUTNOD, <Host node = >
209 000360 DLLDAT SF.DEV, $SDEV, PUTR50,1, <Service device = >, EOL
210
211 000370 DLLDAT SF.CIR, $SCIR, PUTR50,4, <Service circuit = >, EOL
212 000400 DLLDAT SF.PH3, $SFLG, PUTSNT, <Service node version = > EOL
213 000410 DLLDAT SF.PSS, $SPSS, PUTR50,6, <Service password = >, EOL
214
215 000420 DLLDAT SF.HAD, $SHAD, PUTHAD,4, <Hardware address = > EOL
216
217 000430 DLLDAT , F$SEC, PUTFIL,, <Secondary loader file = >, EOL
218 000440 DLLDAT , F$TER, PUTFIL,, <Tertiary loader file = >, EOL
219 000450 DLLDAT , F$LD, PUTFIL,, <Load file = >, EOL
220 000460 DLLDAT , F$DIA, PUTFIL,, <Diagnostics file = >, EOL
221
222 000470 DLLDAT , F$DUM, PUTFIL,, <Dump file = >, EOL
223 000500 DLLDAT $F.DPA, $SDPA, PUTOC1,4, <Dump address = >, EOL
224 000510 DLLDAT $F.DPC, $SDPC, PUTOC2,2, <Dump count = >
225 000520 000000 .WORD 0
226

```

## SYMBOL CROSS REFERENCE

CREF 04.00

| SYMBOL  | VALUE      | REFERENCES                           |
|---------|------------|--------------------------------------|
| ASARG   | 000003     | #4-233 8-436 8-450 8-470             |
| ASFLG   | 000006     | #4-235 8-434                         |
| ASLEN   | = 000010   | #4-236 8-427 8-430                   |
| ASDFS   | 000002     | #4-232 8-433                         |
| ASRTN   | 000004     | #4-234 8-472                         |
| ASSTR   | 000000     | #4-231 8-431                         |
| CHRCOL  | = 000031   | #4-190                               |
| CS.FIL  | = ***** GX | 8-423                                |
| CS.REM  | = ***** GX | 4-272 10-550                         |
| CS.SER  | = ***** GX | 8-417                                |
| C.ADD   | 000010     | #4-118 4-276 4-285 8-419 10-552      |
| C.LEN   | = 000012   | #4-119                               |
| C.NAM   | 000004     | #4-117 4-288 4-289 10-554 10-555     |
| C.STS   | = ***** GX | 4-272 8-417 8-423 10-550             |
| DLLTBL  | 000350 R   | #4-208 8-427                         |
| ERRNOD  | = ***** GX | 4-312                                |
| EXENM   | 000265 R   | #4-194 5-339                         |
| EXENOD  | 001002 R   | 4-265 #5-326                         |
| FILPT   | 000346 R   | #4-203 *8-425 8-444 8-446            |
| FORMAT  | 000217 R   | #4-188 6-375                         |
| FSDIA   | 000012     | #4-162 4-220                         |
| FSDUM   | 000014     | #4-163 4-222                         |
| FSLD    | 000010     | #4-161 4-219                         |
| F\$LEN  | = 000016   | #4-164                               |
| F\$SEC  | 000004     | #4-159 4-217                         |
| F\$TER  | 000006     | #4-160 4-218                         |
| F.CNT   | 000002     | #4-175                               |
| F.DEV   | 000005     | #4-177 9-496                         |
| F.LEN   | = 000043   | #4-178                               |
| F.LNK   | 000000     | #4-174                               |
| GETDLL  | 001246 R   | 4-299 #8-412                         |
| HEADR   | 000000 R   | #4-184 6-357                         |
| HEADR1  | 000033 R   | #4-185 6-357                         |
| HEADR2  | 000075 R   | #4-186 6-357                         |
| HEADR3  | 000142 R   | #4-187 6-357                         |
| HFLAG   | 000332 R   | #4-199 *4-261 4-281 *6-364           |
| HTBL    | 001102 R   | #6-357 6-360                         |
| IN\$CHA | = ***** GX | 6-361 6-373                          |
| I\$SAS  | = *****    | 10-611 10-619                        |
| MS1BLK  | 000334 R   | #4-201 4-284 4-291 5-327 5-334       |
| MXFSP   | = 000036   | #4-170 4-177                         |
| NAMCOL  | = 000017   | #4-189 5-337 8-464                   |
| NODNM   | 000261 R   | #4-193 4-296                         |
| PHASE3  | 000306 R   | #4-195 10-590                        |
| PHASE4  | 000320 R   | #4-196                               |
| PR1HDR  | 001112 R   | 4-283 #6-359                         |
| PUTADR  | 001216 R   | 4-286 5-329 #7-390                   |
| PUTFIL  | 001560 R   | 4-217 4-218 4-219 4-220 4-222 #9-474 |
| PUTHAD  | 002120 R   | 4-215 #10-633                        |
| PUTNOD  | 001614 R   | 4-208 #10-522                        |
| PUTOCT  | 002066 R   | 4-223 #10-616                        |
| PUTOC2  | 002040 R   | 4-224 #10-609                        |

SHOSYS      CREATED BY    MACRO    ON 29-JUN-85 AT 05:28      PAGE 2      M 8  
MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |       |       |       |       |       |
|---------|-------|-------|-------|-------|-------|
| EPRINT  | #3-71 |       |       |       |       |
| ERROR\$ | #3-60 | 3-224 | 3-226 |       |       |
| PRINT   | #3-55 | 3-153 | 3-174 | 3-192 | 3-218 |

```

388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407 000404
408 000404 016700 177370
409 000410 016701 000000G
410 000414 010160 000016
411 000420 020167 177400
412 000424 103403
413 000426 020167 177374
414 000432 101402
415 000434 062716 000002
416 000440 000207

```

```

.SBTTL SETSBS - SET SDB SIZE
+
*** - SETSBS - SET SDB SIZE
REASONABLE CHECKS ARE PERFORMED ON THE CURRENT VALUE. IF THE VALUE IS
NOT IN RANGE, THE C-BIT IS SET.
INPUT:
TEMP = TEMPLATE ADDRESS
$LIMITB = ADDRESS OF LIMITS TABLE
.PNUMB = CURRENT VALUE
.PNUMH = CURRENT VALUE (HIGH ORDER)
OUTPUT:
C-BIT = SUCCESS/FAILURE
VALUE IS STORED IN TEMPLATE.
-
SETSBS:
MOV TEMP,R0 ; POINT AT TEMPLATE
MOV .PNUMB,R1 ; GET THE SIZE OF EACH SDB
MOV R1,C.SBSZ(R0) ; STORE THE SDB SIZE
CMP R1,LOSBSZ ; IS THE VALUE IN RANGE ?
BLO 10$; IF LO, NO - SET C-BIT
CMP R1,HISBSZ ; STILL IN RANGE ?
BLOS 20$; IF LOS, YES - JUST LEAVE
10$: ADD #2,(SP) ; REJECT THE TRANSITION (SETS C-BIT)
20$: RETURN

```

```

136 .SBTTL TPARS STATE TABLES
137 ;
138 ; INITIALIZE TABLES
139 ;
140 ;
141 ISTAT$ CHNSTB, CHNKTB
142 ;
143 ; PROCESS 'CHN$Df'
144 ;
145 STATES$
146 TRANS$ $LAMDA,, $CLERR,ERRCHN,.ERROR
147 STATES$
148 TRANS$ $NUMBR,,SETLO
149 STATES$
150 TRANS$ $EOS,$EXIT
151 TRANS$ <1,>
152 ;
153 STATES$
154 TRANS$ $NUMBR,,SETHI
155 STATES$
156 TRANS$ $EOS,$EXIT
157 ;
158 ; FINAL STATE
159 ;
160 ;
161 STATES$
162

```

192  
 193  
 194  
 195  
 196  
 197  
 198  
 199  
 200  
 201  
 202  
 203  
 204  
 205

```
.SBTIL SETVEC - SETUP THE VECTOR ADDRESS
;+
*** - SETVEC - SETUP THE VECTOR ADDRESS
:
INPUT:
: .PNUMB = THE VECTOR ADDRESS
: .PNUMH = (HIGH ORDER)
: TEMP = START ADDRESS OF THE TEMPLATE
:
OUTPUT:
: THE VECTOR ADDRESS IS MOVED INTO THE TEMPLATE
:-
```

|     |        |        |                |         |        |                  |                            |
|-----|--------|--------|----------------|---------|--------|------------------|----------------------------|
| 206 | 000116 |        |                | SETVEC: | MOV    | TEMP,RO          | ; GET START OF TEMPLATE    |
| 207 | 000116 | 016700 | 177656         |         | MOV    | .PNUMB,C.VEC(RO) | ; STORE THE VECTOR ADDRESS |
| 208 | 000122 | 016760 | 000000G 000006 |         | TST    | .PNUMH           | ; IS IT A WORD VALUE ?     |
| 209 | 000130 | 005767 | 000000G        |         | BNE    | 10\$             | ; IF NE, NO - REJECT       |
| 210 | 000134 | 001010 |                |         | BIT    | #3,.PNUMB        | ; IS IT A MULTIPLE OF FOUR |
| 211 | 000136 | 032767 | 000003 000000G |         | BNE    | 10\$             | ; IF NE, NO - REJECT       |
| 212 | 000144 | 001004 |                |         | CMP    | #774,.PNUMB      | ; IS THE VALUE IN RANGE ?  |
| 213 | 000146 | 022767 | 000774 000000G |         | BHS    | 20\$             | ; IF HIS, YES              |
| 214 | 000154 | 103002 |                |         | ADD    | #2,(SP)          | ; ELSE, REJECT             |
| 215 | 000156 | 062716 | 000002         | 10\$:   |        |                  |                            |
| 216 | 000162 | 000207 |                | 20\$:   | RETURN |                  |                            |

```

153 .SBTTL TPARS STATE TABLES
154 ;
155 ; INITIALIZE TABLES
156 ;
157 ;
158 000116 ISTAT$ CUGSTB, CUGKTB
159 ;
160 ; PROCESS "CUG$DF"
161 ;
162 000116 STATES$
163 000116 TRANS$ $LAMDA,, $CLERR,ERRNAM,.ERROR
164 000116 STATES$; CUG NAME
165 000116 TRANS$ $STRNG,,SETNAM
166 000116 STATES$
167 000116 TRANS$ <','>
168
169 000116 STATES$
170 000116 TRANS$ $LAMDA,, $CLERR,ERRCUG,.ERROR
171 000116 STATES$; GROUP NUMBER
172 000116 TRANS$ $DNUMB,,SETCUG
173 000116 STATES$
174 000116 TRANS$ <','>,CUG2
175 000116 TRANS$ $LAMDA
176 000116 STATES$ CUG2
177 000116 TRANS$ $EOS,$EXIT
178 000116 TRANS$ <','>
179
180 000116 STATES$
181 000116 TRANS$ $LAMDA,, $CLERR,ERRFLG,.ERROR
182 000116 STATES$; FLAGS
183 000116 TRANS$ $NUMBR,,SETFLG
184 000116 STATES$
185 000116 TRANS$ $EOS,$EXIT
186
187 ; FINAL STATE
188 ;
189 000116 STATES$
190

```



```

187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202 000064
203 000064 016700 000000G
204 000070 010701
205 000072 004767 000000G
206 000076 016700 177700
207 000102 010160 000004
208 000106 022767 000003 000000G
209 000114 103002
210 000116 062716
211 000122 000207

 .SBTTL SETNAM - SET UP PROCESS NAME
 ;+
 *** - SETNAM - SETUP PROCESS NAME
 THIS ACTION ROUTINE SETS UP THE PROCESS NAME.
 INPUT:
 .PSTPT = START ADDRESS OF THE PROCESS NAME IN ASCII
 .PSTCN = NUMBER OF CHARACTERS IN THE PROCESS NAME
 OUTPUT:
 C.NAM = RAD50 PROCESS NAME
 -
 SETNAM:
 MOV .PSTPT,R0 ; GET ADDRESS OF ASCII PROCESS NAME
 MOV PC,R1 ; PERIODS ARE ACCEPTABLE
 CALL $CAT5 ; CONVERT PROCESS NAME TO RAD50
 MOV TEMP,R0 ; GET START OF TEMPLATE
 MOV R1,C.NAM(R0) ; STORE PROCESS NAME IN TEMPLATE
 CMP #3,.PSTCN ; IS THE PROCESS NAME TOO BIG ?
 BHS 10$; IF HIS, NO
 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
 10$: RETURN

```

```

120 000002 TRANS $NUMBR,,SETLST ; LINK SERVICE THRESHOLD
121
122 000002 STATES
123 000002 TRANS <','>
124
125 000002 STATES
126 000002 TRANS $NUMBR,,SETRET ; RETRANSMIT FACTOR
127
128 000002 STATES
129 000002 TRANS <','>
130
131 000002 STATES
132 000002 TRANS $NUMBR,,SETSEG ; SEGMENT SIZE
133
134 : EXIT STATE
135 :
136 000002 STATES EXIT
137 000002 TRANS $EOS,$EXIT
138
139 000002 STATES

```

|                  |                     |                     |                   |                   |
|------------------|---------------------|---------------------|-------------------|-------------------|
| CD.LEN= 000022   | DECSTB 000000RG     | 002 NT\$IMS= 000002 | SETLST 000434R    | \$FAIL = 177777   |
| CS.DEC= ***** GX | EXIT 000100R        | 002 NT\$INT= 000004 | SETNDC 000112R    | \$LAMDA= 000000   |
| C.DELF 000016    | MXNDC = 000252      | NT\$RET= 000032     | SETOUT 000214R    | \$NUMBR= 000002   |
| C.DELW 000017    | NM\$ARA= 176000     | NT\$ROU= 000024     | SETRET 000470R    | \$RAD50= 000016   |
| C.INAC 000014    | NM\$NOD= 001777     | NT\$RTR= 000030     | SETSEG 000526R    | \$STRNG= 000004   |
| C.INCT 000012    | NT\$AKD= 000020     | NT\$TSP= 000026     | TEMP 000000R      | \$SUBXP= 000010   |
| C.IPL 000020     | NT\$AKI= 000022     | NM\$ACK= 000011     | \$ALPHA= 000022   | \$TALOC= ***** GX |
| C.LKS 000010     | NT\$CC = 000016     | NM\$HDR= 000007     | \$ANY = 000020    | \$TMDEC 000002RG  |
| C.LST 000021     | NT\$CON= 000000     | NM\$OVR= 000022     | \$BLANK= 000006   | \$\$\$FLG= 177777 |
| C.NDC 000006     | NT\$CTL= 000000     | SETDEF 000310R      | \$DIGIT= 000024   | \$\$\$KEY= 177777 |
| C.OUTT 000013    | NT\$DAT= 000002     | SETDEW 000344R      | \$DNUMB= 000014   | \$\$\$STA= 000000 |
| C.RETF 000015    | NT\$DC = 000012     | SETINA 000252R      | \$EOS = 000012    | .PNUMB= ***** GX  |
| C.SEG 000004     | NT\$DIS= 000014     | SETINC 000156R      | \$ERROR= ***** GX | .PNUMH= ***** GX  |
| C.STS = ***** GX | NT\$DLS= 000006     | SETIPL 000400R      | \$EXIT = 000000   | .TPARS= ***** GX  |
| DECKTB 000000RG  | 003 NT\$ILS= 000010 | SETLKS 000056R      |                   |                   |

. ABS. 000022 000 (RW,I,GBL,ABS,OVR)  
000566 001 (RW,I,LCL,REL,CON)  
\$STATE 000104 002 (RW,D,LCL,REL,CON)  
\$KTAB 000000 003 (RW,D,LCL,REL,CON)  
\$KSTR 000000 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10675 Words ( 42 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSY-11M/PLUS

Elapsed time: 00:00:34.33  
SY:TMPEDEC.V2,[132,134]TMPEDEC/CR/-SP=SY:[1,1]RSXMC.M.SML/ML,[130,110]NETLIB/ML,[132,10]TMPEDEC

```

225 .SBTTL SETPRI - SET THE PRIORITY
226 ;
227 ;*** - SETPRI - SET THE PROCESS PRIORITY
228 ;
229 ; THIS ACTION ROUTINE CHECKS THE DLC PRIORITY TO MAKE SURE IT IS VALID
230 ; AND STORES IT IN THE TEMPLATE.
231 ;
232 ; INPUT:
233 ; .PNUMB = PRIORITY
234 ; .PNUMH = PRIORITY (HIGH ORDER)
235 ;
236 ; OUTPUT:
237 ; THE PRIORITY IS STORED IN C.PRI OF THE TEMPLATE.
238 ;
239 ; -
240
241 000140 SETPRI:
242 000140 016700 177636 MOV TEMP,R0 ; GET START OF TEMPLATE
243 000144 016760 000000G 000010 MOV .PNUMB,C.PRI(R0) ; STORE THE PRIORITY IN THE TEMPLATE
244 ;
245 ; TST .PNUMB ; CHECK THE PRIORITY
246 ; BEQ 10$; IF EQ, OKAY FOR RSX-11M/S
247 ; CMP .PNUMB,#3 ; RSX-11D/IAS PRIORITY?
248 ; BEQ 10$; IF EQ, YES - OKAY
249 000152 000207 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
249 ;
249 ; 10$: RETURN

```

```

56 .SBTTL MACRO DEFINITIONS
57
58 ;****
59 ; LIBRARY MACRO CALLS
60 ;****
61
62 .MCALL FLTDF$
63
64 000000 FLTDF$; DEFINE FILTER BLOCK SYMBOLS
65
66 ;****
67 ; LOCAL MACROS
68 ;****
69
70 .MACRO PRINT TEXT
71 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
72 CALL $PRINT ; PRINT MESSAGE
73 .ENDM PRINT
74
75 .MACRO ERROR$ TEXT
76
77 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
78 ; MESSAGE STRING.
79
80 .IF DIF <TEXT><RO>
81 MOV TEXT,RO ; GET ADDRESS OF ERROR MESSAGE
82 .ENDC
83 CALL $CFERR ; PRINT ERROR MESSAGE
84 .ENDM ERROR$
85
86 .MACRO EPRINT TEXT
87
88 ; PRINT TEXT ON ERROR LUN
89
90 .IF DIF <TEXT><RO>
91 MOV TEXT,RO ; GET ADDRESS OF ASCIZ STRING TO PRINT
92 .ENDC
93 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
94 .ENDM EPRINT
95

```

## SYMBOL CROSS REFERENCE

CREF 04.00

| SYMBOL  | VALUE      | REFERENCES                                        |
|---------|------------|---------------------------------------------------|
| HEADR1  | 000112 R   | #5-105 7-202                                      |
| MONITR  | 000267 R   | #5-114 5-115 8-278                                |
| MONL    | = 000007   | #5-115 8-277                                      |
| MSGBLK  | 000316 R   | #5-125 8-266 9-299                                |
| MSKLEN  | = 000100   | #6-139 10-340                                     |
| NOINFO  | 000207 R   | #5-107 7-246                                      |
| CFF     | 000264 R   | #5-112 5-113 9-301                                |
| OFFL    | = 000003   | #5-113 9-300                                      |
| OM.CON  | = ***** GX | 7-217                                             |
| OM.FIL  | = ***** GX | 7-220                                             |
| ON      | = 000262 R | #5-110 5-111 9-305                                |
| ONL     | = 000002   | #5-111 9-304                                      |
| PERIOD  | = 000056   | #6-153 10-339                                     |
| PREVNT  | 000314     | #5-123 *10-351 *10-358 *10-367 11-396             |
| PRTEVT  | 001014     | 7-242 #10-325                                     |
| PRTSNK  | 000620     | 7-223 #8-266                                      |
| PRTSTA  | 000730     | 7-227 #9-299                                      |
| SNKMSG  | 000166     | #5-106 8-280                                      |
| SNKTBL  | 000322     | #5-129 7-212                                      |
| STAMSG  | 000230     | #5-108 9-307                                      |
| STATE   | 000315 R   | #5-124 *7-195 9-302                               |
| TAB     | = 000011   | #6-150 10-332                                     |
| TMPLOC  | = 000004   | #6-161 6-169                                      |
| WC.LOG  | = ***** GX | 7-200 7-210 7-247                                 |
| \$BUFF  | = ***** GX | 7-203 7-206 8-279 8-282 9-306 9-309 10-331 10-376 |
| \$CBDMG | = ***** GX | 10-338 10-348 11-399                              |
| \$CFERR | = ***** GX | 7-191                                             |
| \$EDMSG | = ***** GX | 7-205 8-281 9-308                                 |
| \$GTBUF | = ***** GX | 7-204                                             |
| \$OPRND | = ***** GX | 7-217 7-220                                       |
| \$PRINT | = ***** GX | 7-206 7-246 8-282 9-309 10-329 10-376             |
| \$SHLOG | 000332 RG  | #7-187                                            |
| \$TMLST | = ***** GX | 7-188 7-229                                       |
| \$WILD  | = ***** GX | 7-200 7-210 7-247                                 |

```

403 .SBTIL PRITCHN - PRINT DTE CHANNEL LIST
404
405 ** PRITCHN - PRINT DTE CHANNEL LIST
406
407 INPUTS R5 = ADDRESS OF DTE TEMPLATE
408
409 OUTPUT THE DTE CHANNEL LIST IS OUTPUT.
410
411 REGISTERS R5 IS PRESERVED.
412
413
414 PRITCHN:
415 002560 MOV R5,R4 ; COPY DTE POINTER
416 002562 MOV #CHNHDR,LINHDR ; FIRST LINE GETS LINE HFADER
417 002570 MOV #CHNBUF,OBUF ; INITIALIZE BUFFER POINTER
418
419 002576 10$: MOV (R4),R4 ; NEXT TEMPLATE
420 002600 BEQ 60$; IF EQ, NO MORE CHANNELS
421 002602 122764 000000G 000000G CMPB #CS.CHN,C.STS(R4) ; CHECK FOR CHANNEL TEMPLATE
422 002610 BNE 60$; IF NE, NO MORE CHANNELS
423 002612 012701 000255' MOV #FMTRNM,R1 ; ASSUME MULTIPLE CHANNELS
424 002616 032764 000000G 000000G BIT #CH$HI,C.STS(R4) ; TEST FOR HIGH CHANNEL
425 002624 001002 BNE 20$; IF NE, HIGH CHANNEL PRESENT
426 002626 012701 000250' MOV #FMTRN1,R1 ; ELSE SINGLE CHANNEL
427
428 002632 016700 175146 20$: MOV OBUF,R0 ; GET OUTPUT BUFFER POINTER
429 002636 020027 000034' CMP R0,#CHNBUF ; ARE WE STARTING A LINE ?
430 002642 001006 BNE 40$; IF NE, NO
431 002644 016702 175132 MOV LINHDR,R2 ; POINT TO LINE HEADER
432 002650 112220 30$: MOV (R2)+,(R0)+ ; MOVE A CHARACTER
433 002652 001376 BNE 30$; UNTIL ZERO BYTE
434 002654 005300 DEC R0 ; BACK OVER ZERO BYTE
435 002656 005201 INC R1 ; NO COMMA SEPARATOR
436
437 002660 010402 40$: MOV R4,R2 ; COPY TEMPLATE POINTER
438 002662 062702 000000G ADD #C$HLO,R2 ; POINT TO ARGUMENT BLOCK
439 002666 004767 000000G CALL $EDMSG ; FORMAT OUTPUT
440 002672 020027 0C0117' CMP R0,#CHN$END-9. ; ROOM FOR MORE IN BUFFER ?
441 002676 002422 BLT 50$; IF LT, YES
442 002700 011402 MOV (R4),R2 ; POINT AHEAD A TEMPLATE
443 002702 001407 BEQ 45$; IF EQ, NO MORE CHANNELS
444 002704 122762 000000G 000000G CMPB #CS.CHN,C.STS(R2) ; MORE CHANNELS ?
445 002712 001003 BNE 45$; BR IF NO
446 002714 112720 000054 MOV #<','>,(R0)+ ; ELSE INSERT TRAILING
447 002720 105020 CLRB (R0)+ ; COMMA
448
449 002722 45$: PRINT #CHNBUF ; ELSC OUTPUT THE BUFFER
450 002732 012700 000034' MOV #CHNBUF,R0 ; RESET BUFFER POINTER
451 002736 012767 001101' 175036 MOV #CHNBLK,LINHDR ; LEADING BLANKS HEREAFTER
452 002744 010067 175034 50$: MOV R0,OBUF ; UPDATE BUFFER POINTER
453 002750 000712 BR 10$; LOOK FOR MORE
454
455 002752 026727 175026 000034' 60$: CMP OBUF,#CHNBUF ; SEE IF ANY DATA IN BUFFER
456 002760 001404 BEQ 70$; BR IF NONE
457 002762 PRINT #CHNBUF ; OUTPUT DATA
458 002772 000207 70$: RETURN
459

```

869 004450 012701 000300'  
 870 004454 010502  
 871 004456 062702 000000G  
 872 004462 004767 000000G  
 873 004466 162700 000034'  
 874 004472 010023  
 875 004474 012723 000034'  
 876  
 877  
 878  
 879 004500 011505  
 880 004502  
 881 004516  
 882 004532  
 883 004546  
 884  
 885  
 886  
 887 004562 011505  
 888 004564  
 889 004600  
 890  
 891  
 892  
 893 004614 012703 001532'  
 894 004620 004767 177206  
 895 004624 000167 177426  
 896  
 897 004630 000207  
 898

```

MOV #FMT2R5,R1 ; ELSE USE RAD50 TASK NAME
MOV R5,R2 ; COPY TEMPLATE POINTER
ADD #D$STSK,R2 ; POINT TO TASK NAME
70$: CALL $EDMSG ; BUILD OUTPUT STRING
SUB #CHNBUF,R0 ; GET LENGTH OF TASK NUMBER/NAME STRING
MOV R0,(R3)+ ; STUFF IN ARG BLOCK
MOV #CHNBUF,(R3)+ ; STRING ADDRESS

;
; OPTIONAL PARAMETERS: DTE NUMBER, SUBADDRESS RANGE AND GROUP
;
MOV (R5),R5 ; POINT TO DSA$DF BLOCK
OPTION D$ANUM,5,CS.NUM,DS.NUM ; DTE NUMBER
OPTION D$ACUG,2,DA$CUG,DS.CUG ; GROUP ID
OPTION D$ASLO,1,CS.SLO,DS.CLO ; SUBADDRESS LOW
OPTION D$ASHI,1,CS.SHI,DS.CHI ; SUBADDRESS HIGH

;
; CALL MASK AND CALL VALUE
;
80$: MOV (R5),R5 ; POINT TO DSC$DF
OPTION D$CMSK,11,,CS.MSK,DS.MSK ; CALL MASK
OPTION D$CVAL,11,,CS.VAL,DS.VAL ; CALL VALUE

;
; SHOW CHARACTERISTICS
;
90$: MOV #DSTFTR3 ; POINT TO FORMATTING TABLE
CALL PRINT ; OUTPUT DATA
JMP 10$; CONTINUE

10$: RETURN

```



```

53 .SBTIL MACRO DEFINITIONS
54
55 .MCALL LLCTP$
56
57 000000 LLCTP$ LIST ; DEFINE LLC$DF TEMPLATE OFFSETS
 ;
 ; DEFINE LLC TEMPLATE
 ;
000000 .A$ECT
000000 .=0
000002 .BLKW 1 ; C.LNK
000004 .BLKW 1 ; C.STS
000006 C.NAM: .BLKW 1 ; LLC PROCESS NAME
000010 C.FLG: .BLKW 1 ; LLC FLAGS WORD
000011 C.PRI: .BLKB 1 ; LLC PROCESS PRIORITY
000012 C.LIN: .BLKB 1 ; NUMBER OF LINES ASSIGNED TO THIS LLC
000014 C.DEV: .BLKW 1 ; PSEUDO DEVICE NAME FOR THIS LLC
000015 C.NLT: .BLKB 1 ; NUMBER OF LINE TABLE TO ALLOCATE
000016 C.EXT: .BLKB 1 ; PROCESS EXTENSION IN BLOCKS
000020 C.CTM: .BLKW 1 ; COUNTER TIMER
000020 CL.LEN=.
000012 .P$ECT
000013 C.NLTD = 12 ; NLT FOR DDM/DLC PROCESSES
000013 C.NCTD = 13 ; NCT FOR DDM/DLC PROCESSES
58 ;****
59 ; LOCAL MACROS
60 ;****
61
62 .MACRO PRINT TEXT
63 .IF DIF <TEXT><R0>
64 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
65 .ENDC
66 CALL $PRINT ; PRINT MESSAGE
67 .ENDM PRINT
68
69 .MACRO ERROR$ TEXT
70 ;
71 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
72 ; MESSAGE STRING.
73 ;
74 .IF DIF <TEXT><R0>
75 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
76 .ENDC
77 CALL $CFERR ; PRINT ERROR MESSAGE
78 .ENDM ERROR$
79
80 .MACRO EPRINT TEXT
81 ;
82 ; PRINT TEXT ON ERROR LUN
83 ;
84 .IF DIF <TEXT><R0>
85 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
86 .ENDC
87 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
88 .ENDM EPRINT
89
90 ;****

```

```

227 ; ARGUMENT BLOCK FORMAT FOR DLL FORMAT CONTROL TABLE
228 ;
229 000043 .ASECT
230 000000 . = 0
231 000000 ASSTR: .BLKW 1 ; address of ASCIZ string
232 000002 ASOFS: .BLKB 1 ; Sxxx/Fxxx offset
233 000003 ASARG: .BLKB 1 ; argument for fmt routine, hi bit = EOL before printing
234 000004 ASRTN: .BLKW 1 ; address of formatting routine
235 000006 ASFLG: .BLKW 1 ; SF.xxx flag in SER$DF - 0= unconditional print
236 000010 ASLEN =
237 000522 .PSECT
238
239 .SBTTL $SHREM - SHOW REMOTE CHARACTERISTICS
240
241 ;+
242 *** - $SHREM - SHOW REMOTE CHARACTERISTICS
243
244 INPUT:
245 $NODNA = THE NODE NAME TO DISPLAY
246 $WILD = WILD CARD SPECIFICATIONS
247
248 OUTPUT:
249 THE REMOTE CHARACTERISTICS ARE DISPLAYED.
250 RO-R5 = DESTROYED
251
252 ; -
253
254 000522 $SHREM::
255 000522 032767 176000 000000G BIT #C1777,$NODAD ; AREA SPECIFIED ?
256 000530 001006 BNE 2$; IF NE, YES
257 000532 016700 000000G MOV $NTADD,R0 ; ELSE FOLD ...
258 000536 042700 001777 BIC #1777,R0 ; ... IN ...
259 000542 050067 000000G BIS RO,$NODAD ; ... EXECUTOR AREA
260 000546 005000 2$: CLR RO ; SUCCESS INDICATOR
261 000550 005067 CLR HFLAG ; HEADER PRINT FLAG
262 000554 032767 000000G BIT #WC.NOD,$WILD ; IS THIS FOR KNOWN NODES ?
263 000562 001404 BEQ 5$; IF EQ, NO
264 000564 004767 000322 CALL PRIHDR ; ELSE, PRINT THE HEADER NOW
265 000570 004767 000206 CALL EXENOD ; AND PRINT THE EXECUTOR NODE INFO
266 000574 012705 000000G 5$: MOV #$TMLST,R5 ; GET START OF TEMPLATE LIST
267
268 ; SEARCH FOR THE REMOTE TEMPLATE
269
270 000600 011505 10$: MOV (R5),R5 ; GET NEXT TEMPLATE
271 000602 001463 BEQ 40$; IF EQ, END OF TEMPLATES
272 000604 122765 000000G 000000G CMPB #CS.REM,C.STS(R5) ; IS THIS A REMOTE TEMPLATE ?
273 000612 001372 BNE 10$; IF NE, NO .. KEEP LOOKING
274 000614 032767 000000G 000000G BIT #WC.NOD,$WILD ; DO IT FOR ALL NODES ?
275 000622 001004 BNE 20$; IF NE, S
276 000624 026765 000000G 000010 CMP $NODAD,C.ADD(R5) ; IS THIS THE RIGHT TEMPLATE ?
277 000632 001362 BNE 10$; IF NE, NO .. KEEP LOOKING
278
279 ; DISPLAY THE REMOTE CHARACTERISTICS
280
281 000634 005767 177472 20$: TST HFLAG ; PRINT HEADER ?
282 000640 001002 BNE 25$; IF NE, NO
283 000642 004767 000244 CALL PRIHDR ; ELSE, PRINT THE HEADER

```

## SYMBOL CROSS REFERENCE

CREF 04.00

## SYMBOL VALUE REFERENCES

|           |        |        |        |         |        |         |        |        |        |        |        |        |  |  |
|-----------|--------|--------|--------|---------|--------|---------|--------|--------|--------|--------|--------|--------|--|--|
| PUTR50    | 002004 | R      | 4-209  | 4-211   | 4-213  | #10-575 | 10-637 |        |        |        |        |        |  |  |
| PUTSNT    | 002024 | R      | 4-212  | #10-590 |        |         |        |        |        |        |        |        |  |  |
| P3NOD     | 000246 | R      | #4-191 |         |        |         |        |        |        |        |        |        |  |  |
| P4NOD     | 000252 | R      | #4-192 | 4-287   | 5-330  |         |        |        |        |        |        |        |  |  |
| R\$\$EIS  | =      | *****  | 10-611 | 10-619  |        |         |        |        |        |        |        |        |  |  |
| R\$\$I1D  | =      | *****  | 10-611 | 10-619  |        |         |        |        |        |        |        |        |  |  |
| SCR       | 000334 | R      | #4-200 | 10-612  |        |         |        |        |        |        |        |        |  |  |
| SERPT     | 000344 | R      | #4-202 | *8-421  | 8-438  | 10-523  | 10-575 | 10-609 | 10-616 |        |        |        |  |  |
| SF.ADD    | =      | 000001 | #4-142 |         |        |         |        |        |        |        |        |        |  |  |
| SF.CIR    | =      | 000010 | #4-145 | 4-211   |        |         |        |        |        |        |        |        |  |  |
| SF.DEV    | =      | 000020 | #4-146 | 4-209   |        |         |        |        |        |        |        |        |  |  |
| SF.DPA    | =      | 000100 | #4-148 | 4-223   |        |         |        |        |        |        |        |        |  |  |
| SF.DPC    | =      | 000200 | #4-149 | 4-224   |        |         |        |        |        |        |        |        |  |  |
| SF.HAD    | =      | 000004 | #4-144 | 4-215   |        |         |        |        |        |        |        |        |  |  |
| SF.HST    | =      | 000002 | #4-143 | 4-208   |        |         |        |        |        |        |        |        |  |  |
| SF.PH3    | =      | 000400 | #4-150 | 4-212   |        |         |        |        |        |        |        |        |  |  |
| SF.PSS    | =      | 000040 | #4-147 | 4-213   |        |         |        |        |        |        |        |        |  |  |
| S\$ADD    | 000006 |        | #4-129 | 8-419   |        |         |        |        |        |        |        |        |  |  |
| S\$CIR    | 000022 |        | #4-132 | 4-211   |        |         |        |        |        |        |        |        |  |  |
| S\$DEV    | 000032 |        | #4-133 | 4-209   |        |         |        |        |        |        |        |        |  |  |
| S\$DPA    | 000050 |        | #4-135 | 4-223   |        |         |        |        |        |        |        |        |  |  |
| S\$DPC    | 000054 |        | #4-136 | 4-224   |        |         |        |        |        |        |        |        |  |  |
| S\$FLG    | 000004 |        | #4-128 | 4-212   | 8-440  |         |        |        |        |        |        |        |  |  |
| S\$HAD    | 000012 |        | #4-131 | 4-215   |        |         |        |        |        |        |        |        |  |  |
| S\$HST    | 000010 |        | #4-130 | 4-208   |        |         |        |        |        |        |        |        |  |  |
| S\$LEN    | =      | 000056 | #4-137 |         |        |         |        |        |        |        |        |        |  |  |
| S\$PSS    | 000034 |        | #4-134 | 4-213   |        |         |        |        |        |        |        |        |  |  |
| WC.NOD    | =      | *****  | GX     | 4-262   | 4-274  | 4-306   | 6-365  |        |        |        |        |        |  |  |
| \$BUFF    | =      | *****  | GX     | 4-290   | 4-294  | 4-298   | 5-333  | 5-337  | 5-341  | 6-368  | 6-372  | 8-428  |  |  |
|           |        |        |        | 8-452   | 8-454  | 8-455   | 8-457  | 8-464  | 8-476  | 8-478  |        |        |  |  |
| \$CBDMG   | =      | *****  | GX     | 10-534  | 10-539 |         |        |        |        |        |        |        |  |  |
| \$CBTA3   | =      | *****  | GX     | 10-621  |        |         |        |        |        |        |        |        |  |  |
| \$CFERR   | =      | *****  | GX     | 4-312   |        |         |        |        |        |        |        |        |  |  |
| \$CSTA    | =      | *****  | GX     | 10-559  | 10-551 |         |        |        |        |        |        |        |  |  |
| \$CSTA1   | =      | *****  | GX     | 10-578  |        |         |        |        |        |        |        |        |  |  |
| \$EDMSG   | =      | *****  | GX     | 4-292   | 4-297  | 5-335   | 5-340  | 6-371  |        |        |        |        |  |  |
| \$GTBUF   | =      | *****  | GX     | 6-370   |        |         |        |        |        |        |        |        |  |  |
| \$INFO    | =      | *****  | GX     | 6-361   | 6-373  |         |        |        |        |        |        |        |  |  |
| \$NODAD   | =      | *****  | GX     | 4-255   | *4-259 | 4-276   |        |        |        |        |        |        |  |  |
| \$NTADD   | =      | *****  | GX     | 4-257   | 5-328  | 10-542  |        |        |        |        |        |        |  |  |
| \$NTNAM   | =      | *****  | GX     | 5-331   | 5-332  | 10-544  |        |        |        |        |        |        |  |  |
| \$PRINT   | =      | *****  | GX     | 4-298   | 5-341  | 6-372   | 10-545 | 8-454  | 8-478  |        |        |        |  |  |
| \$SAVAL   | =      | *****  | GX     | 8-412   |        |         |        |        |        |        |        |        |  |  |
| \$SHREM   | =      | 000522 | RG     | #4-254  |        |         |        |        |        |        |        |        |  |  |
| \$TMLST   | =      | *****  | GX     | 4-266   | 8-413  | 10-541  |        |        |        |        |        |        |  |  |
| \$WILD    | =      | *****  | GX     | 4-262   | 4-274  | 4-306   | 6-365  |        |        |        |        |        |  |  |
| \$\$\$    | =      | 000341 | R      | #4-208  | 4-208  | #4-209  | 4-209  | #4-211 | 4-211  | #4-212 | 4-212  | #4-213 |  |  |
|           |        |        |        | 4-213   | #4-215 | 4-215   | #4-217 | 4-217  | #4-218 | 4-218  | #4-219 | 4-219  |  |  |
|           |        |        |        | 4-220   | 4-220  | #4-222  | 4-222  | #4-223 | 4-223  | #4-224 | 4-224  |        |  |  |
| \$\$\$EOL | =      | 000000 |        | #4-208  | 4-208  | #4-209  | 4-209  | #4-211 | 4-211  | #4-212 | 4-212  | #4-213 |  |  |
|           |        |        |        | #4-212  | 4-212  | #4-213  | 4-213  | #4-215 | 4-215  | #4-216 | 4-216  | #4-217 |  |  |
|           |        |        |        | #4-217  | 4-217  | #4-217  | 4-217  | #4-218 | 4-218  | #4-219 | 4-219  | #4-220 |  |  |

\*\*FILE\*\*ID\*\*TMPBUF

|            |      |      |          |          |            |    |            |      |
|------------|------|------|----------|----------|------------|----|------------|------|
| TTTTTTTTTT | MM   | MM   | PPPPPPPP | 88888888 | UU         | UU | FFFFFFFFFF |      |
| TTTTTTTTTT | MM   | MM   | PPPPPPPP | 88888888 | UU         | UU | FFFFFFFFFF |      |
| TT         | MMMM | MMMM | PP       | 88       | 88         | UU | FF         |      |
| TT         | MMMM | MMMM | PP       | 88       | 88         | UU | FF         |      |
| TT         | MM   | MM   | PP       | 88       | 88         | UU | FF         |      |
| TT         | MM   | MM   | PP       | 88       | 88         | UU | FF         |      |
| TT         | MM   | MM   | PPPPPPPP | 88888888 | UU         | UU | FFFFFFFFFF |      |
| TT         | MM   | MM   | PPPPPPPP | 88888888 | UU         | UU | FFFFFFFFFF |      |
| TT         | MM   | MM   | PP       | 88       | 88         | UU | FF         |      |
| TT         | MM   | MM   | PP       | 88       | 88         | UU | FF         |      |
| TT         | MM   | MM   | PP       | 88       | 88         | UU | FF         | .... |
| TT         | MM   | MM   | PP       | 88       | 88         | UU | FF         | .... |
| TT         | MM   | MM   | PP       | 88888888 | UUUUUUUUUU | UU | FF         | .... |
| TT         | MM   | MM   | PP       | 88888888 | UUUUUUUUUU | UU | FF         | .... |

|            |          |            |
|------------|----------|------------|
| LL         | SSSSSSSS | TTTTTTTTTT |
| LL         | SSSSSSSS | TTTTTTTTTT |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SSSSSS   | TT         |
| LL         | SSSSSS   | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LLLLLLLLLL | SSSSSSSS | TT         |
| LLLLLLLLLL | SSSSSSSS | TT         |

418  
 419  
 420  
 421  
 422  
 423  
 424  
 425  
 426  
 427  
 428  
 429  
 430  
 431  
 432  
 433  
 434  
 435  
 436  
 437  
 438  
 439  
 440  
 441  
 442  
 443  
 444  
 445  
 446  
 447  
 448

000442  
 016700 177332  
 016701 000000G  
 010160 000020  
 010167 000000G  
 020167 177342  
 103404  
 026767 000000G 177334  
 101402  
 062716 000002  
 000207

```

.SBTTL SETTHR - SET RDB THRESHOLD
+
*** - SETTHR - SET RDB THRESHOLD
REASONABLE CHECKS ARE PERFORMED ON THE CURRENT VALUE. IF THE VALUE IS
NOT IN RANGE, THE C-BIT IS SET.
INPUT:
TEMP = TEMPLATE ADDRESS
$LIMITB = ADDRESS OF LIMITS TABLE
.PNUMB = CURRENT VALUE
.PNUMH = CURRENT VALUE (HIGH ORDER)
OUTPUT:
C-BIT = SUCCESS/FAILURE
VALUE IS STORED IN TEMPLATE.
-
SETTHR:
MOV TEMP,R0 ; POINT AT TEMPLATE
MOV .PNUMB,R1 ; GET THE RDB THRESHOLD COUNT
MOV R1,C.THSH(R0) ; STORE THE RDB THRESHOLD
MOV R1,$THRS ; SAVE THE RDB THRESHOLD COUNT
CMP R1,LOTHSH ; IS THE VALUE IN RANGE ?
BLO 10$; IF LO, NO - SET C-BIT
CMP .PNUMB,HITHSH ; STILL IN RANGE ?
BLOS 20$; IF LOS, YES - JUST LEAVE
ADD #2,(SP) ; REJECT THE TRANSITION (SETS C-BIT)
10$: RETURN
20$:
:::

```

164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179  
180  
181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200

.SBTTL \$TMCHN - BUILD CHANNEL TEMPLATE

\*\*\* - \$TMCHN - BUILD CHANNEL TEMPLATE

THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.

INPUT:

R3-R5 - TPARS REGISTERS

OUTPUT:

C-BIT = SUCCESS/FAILURE

THE CHANNEL TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.

\$TMCHN::

```

MOV R5, -(SP) ; SAVE R5
MOV #CHLEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
MOV R0, TEMP ; SAVE TEMPLATE ADDRESS
TST (R0)+ ; POINT TO C, STS
MOV R0, .FLAGS ; SAVE FOR OPTION SETTING
MOV #CS.CHN, (R0) ; INDICATE THAT THIS IS A CHN$DF TEMPLATE
CLR .ERROR ; NO ERROR DETAIL YET
CLR R1 ; IGNORE BLANKS
MOV #CHNKTBL, R2 ; KEYWORD TABLE
MOV #CHNSTB, R5 ; STATE TABLE
CALL .TPARS ; PARSE THE REST OF THE LINE
BCC 20$; NORMAL RETURN IF NO ERROR
MOV .ERROR, R0 ; CHECK FOR ERROR DETAIL TEXT
BEQ 15$; IF EQ, NO DETAIL TEXT
ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
MOV #-2, $ERROR ; INDICATE MESSAGE PRINTED
JNC $ERROR ; INDICATE SYNTAX ERROR
MOV (SP)+, R5 ; RESTORE R5
RETURN

```

177776 000000G

15\$:  
20\$:

```

218 .SBTTL SETCSR - SET THE CSR ADDRESS
219 +
220 : *** - SETCSR - SET THE CSR ADDRESS
221 :
222 : INPUT:
223 : .PNUMB = CSR ADDRESS
224 : .PNUMH = CSR ADDRESS (HIGH ORDER)
225 : TEMP = TEMPLATE ADDRESS
226 :
227 : OUTPUT:
228 : THE CSR ADDRESS IS STORED IN C.CSR OF THE TEMPLATE.
229 :
230 : -
231
232 SETCSR:
233 000164 016700 177610 MOV TEMP,R0 ; GET START OF TEMPLATE
234 000170 016760 000000G 000010 MOV .PNUMB,C.CSR(R0) ; STORE THE CSR ADDRESS IN THE TEMPLATE
235 000176 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
236 000202 001004 BNE 10$; IF NE, NO - REJECT
237 000204 022767 160000 000000G CMP #160000,.PNUMB ; IS THE CSR ADDRESS IN RANGE ?
238 000212 101402 BLOS 20$; IF LOS, YES
239 000214 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
240 000220 000207 20$: RETURN

```

```

192 .SBTTL $TMCUG - BUILD CUG TEMPLATE
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207 $TMCUG::
208 MOV R5, -(SP) ; SAVE R5
209 MOV #C.LEN,R1 ; GET LENGTH OF NEEDED ALLOCATION
210 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
211 MOV R0,TEMP ; SAVE TEMPLATE ADDRESS
212 TST (R0)+ ; POINT TO C.STS
213 MOV R0,FLAGS ; SAVE FOR OPTION SETTING
214 MOV #CS.CUG,(R0) ; INDICATE THAT THIS IS A CUG$DF TEMPLATE
215 CLR .ERROR ; NO ERROR DETAIL YET
216 ADD #G$LENX,$BYTMN ; UPDATE MINIMUM BYTE ALLOCATION
217 CLR R1 ; IGNORE BLANKS
218 MOV #CUGKTB,R2 ; KEYWORD TABLE
219 MOV #CUGSTB,R5 ; STATE TABLE
220 CALL TPARS ; PARSE THE REST OF THE LINE
221 BCC 20$; NORMAL RETURN IF NO ERROR
222 MOV .ERROR,R0 ; CHECK FOR ERROR DETAIL TEXT
223 BEQ 15$; IF EQ, NO DETAIL TEXT
224 MOV R0 ; PRINT ACTION ROUTINE ERROR
225 MOV #-2,$ERROR ; SUPPRESS SYNTAX MESSAGE
226 INC $ERROR ; INDICATE SYNTAX ERROR
227 MOV (SP)+,R5 ; RESTORE R5
228 RETURN
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248

```

000116 010546  
 000120 012701 000013  
 000124 004767 000000G  
 000130 010067 000000G  
 000134 005720  
 000136 010067 000000G  
 000142 012710 000000G  
 000146 005067 000000G  
 000152 062767 000000G 000000G  
 000160 005001  
 000162 012702 000000'  
 000166 012705 000000'  
 000172 004767 000000G  
 000176 103012  
 000200 016700 000000G  
 000204 001405  
 000206  
 000212 012767 177776 000000G  
 000220 005267 000000G  
 000224 012605  
 000226 000207

15\$: INC \$ERROR  
 20\$: MOV (SP)+,R5

```

 .SBTTL SETNAM - SET CUG NAME
 .SBTTL SETCUG - SET GROUP NUMBER
 .SBTTL SETFLG - SET FLAGS BYTE

 ** SETNAM - SET CUG NAME
 ** SETCUG - SET GROUP NUMBER
 ** SETFLG - SET FLAGS BYTE

 INPUTS .TPARS VARIABLES.

 OUTPUT VALUE SET IN TEMPLATE.

```

000230 SETNAM: \$GTR50 #C.NAM,#2  
 000244 SETCUG: \$GTNUM #C.CUG,,#9999.  
 000264 SETFLG: \$GTNMB #C.FLG,,#CS.FLG



213  
 214  
 215  
 216  
 217  
 218  
 219  
 220  
 221  
 222  
 223  
 224  
 225  
 226  
 227  
 228  
 229

000124  
 000124 016700 177652  
 000130 016760 177644 000006  
 000136 000207

```
.SBTTL SETFLG - SETUP THE FLAGS WORD
;+
;*** - SETFLG - SETUP THE FLAGS WORD
;
;INPUT:
; FLAG = THE FLAGS WORD
; TEMP = START ADDRESS OF THE TEMPLATE
;
;OUTPUT:
; THE FLAGS WORD IS MOVED INTO THE TEMPLATE
;-
SETFLG:
 MOV TEMP,R0 ; GET START OF TEMPLATE
 MOV FLAG,C.FLG(R0) ; STORE THE FLAGS WORD
 RETURN
```

```

141 .SBTTL $TMDEC - SETUP DEC TEMPLATE
142 ;+
143 *** - $TMDEC - SETUP DEC TEMPLATE
144 ;
145 INPUT:
146 R3-R5 - TPARS REGISTERS
147 ;
148 OUTPUT:
149 C-BIT = SUCCESS/FAILURE
150 IF SUCCESS, THE DEC TEMPLATE IS STORED IN THE END OF TASK BUFFER.
151 ;
152 :-
153
154 000002 $TMDEC::
155 000002 010546 MOV R5, -(SP) ; SAVE R5
156 000004 012701 000022 MOV #CD.LEN, R1 ; GET LENGTH OF ALLOCATION
157 000010 004767 000000G CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
158 000014 010067 177760 MOV R0, TEMP ; STORE TEMPLATE ADDRESS
159 000020 012760 000000G 000000G MOV #CS.DEC, C.STS(R0) ; INDICATE THAT THIS IS AN DEC$DF TEMPLATE
160 000026 005001 CLR R1 ; IGNORE BLANKS
161 000030 012702 000000' MOV #DECKTB, R2 ; KEYWORD TABLE
162 000034 012705 000000' MOV #DECSTB, R5 ; STATE TABLE
163 000040 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
164 000044 103002 BCC 20$; NORMAL RETURN IF NO ERROR
165 000046 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
166 000052 012605 20$: MOV (SP)+, R5 ; RESTORE R5
167 000054 000207 RETURN

```

TMPDEC      CREATED BY MACRO ON 29-JUN-85 AT 05:31      PAGE 1      N 15  
 SYMBOL CROSS REFERENCE      CREF      04.00

| SYMBOL  | VALUE       | REFERENCES                                                                                         |
|---------|-------------|----------------------------------------------------------------------------------------------------|
| CD.LEN  | = 000022    | #4-52      6-156                                                                                   |
| CS.DEC  | = ***** GX  | 6-159                                                                                              |
| C.DELF  | 000016      | #4-52      *12-317                                                                                 |
| C.DELW  | 000017      | #4-52      *13-342                                                                                 |
| C.INAC  | 000014      | #4-52      *11-291                                                                                 |
| C.INCT  | 000012      | #4-52      *9-229                                                                                  |
| C.IPL   | 000020      | #4-52      *14-368                                                                                 |
| C.LKS   | 000010      | #4-52      *7-186                                                                                  |
| C.LST   | 000021      | #4-52      *15-393                                                                                 |
| C.NDC   | 000006      | #4-52      *8-214                                                                                  |
| C.QUIT  | 000013      | #4-52      *10-265                                                                                 |
| C.REF   | 000015      | #4-52      *16-418                                                                                 |
| C.SEG   | 000004      | #4-52      *17-443                                                                                 |
| C.STS   | = ***** GX  | *6-159                                                                                             |
| DECKTB  | 000000 RG   | #5-69      6-161                                                                                   |
| DECSTB  | 000000 RG   | #5-69      6-162                                                                                   |
| MXNDC   | = 000252    | #4-57      8-212                                                                                   |
| N\$DOVR | = 000022    | 17-445                                                                                             |
| SETDEF  | 000310 R    | #12-315                                                                                            |
| SETDEW  | 000344 R    | #13-340                                                                                            |
| SETINA  | 000252 R    | #11-289                                                                                            |
| SETINC  | 000156 R    | #9-237                                                                                             |
| SETIPL  | 000400 R    | #14-366                                                                                            |
| SETLKS  | 000056 R    | #7-184                                                                                             |
| SETLST  | 000434 R    | #15-391                                                                                            |
| SETNDC  | 000112 R    | #8-210                                                                                             |
| SETOUT  | 000214 R    | #10-263                                                                                            |
| SETRET  | 000470 R    | #16-416                                                                                            |
| SETSEG  | 000526 R    | #17-441                                                                                            |
| TEMP    | 000000 R    | #4-61      *6-158      7-185      8-211      9-238      10-264      11-290      12-316      13-341 |
|         |             | 14-367      15-392      16-417      17-442                                                         |
| \$ALPHA | = 000022    | #5-69                                                                                              |
| \$ANY   | = 000020    | #5-69                                                                                              |
| \$BLANK | = 000006    | #5-69                                                                                              |
| \$DIGIT | = 000024    | #5-69                                                                                              |
| \$DNUMB | = 000014    | #5-69                                                                                              |
| \$EOS   | = 000012    | #5-69                                                                                              |
| \$ERROR | = ***** GX  | *6-165                                                                                             |
| \$EXIT  | = 000000    | #5-69                                                                                              |
| \$FAIL  | = 177777    | #5-69                                                                                              |
| \$GPRM  | = *****     | 5-69                                                                                               |
| \$LAMDA | = 000000    | #5-69                                                                                              |
| \$NUMBR | = 000002    | #5-69                                                                                              |
| \$RAD50 | = 000016    | #5-69                                                                                              |
| \$RONLY | = *****     | 5-69      5-69                                                                                     |
| \$STRNG | = 000004    | #5-69                                                                                              |
| \$SUBXP | = 000010    | #5-69                                                                                              |
| \$TALOC | = ***** GX  | 6-157                                                                                              |
| \$TMDEC | = 000002 RG | #6-154                                                                                             |
| \$SFLG  | = 177777    | #5-69                                                                                              |
| \$SKEY  | = 177777    | #5-69                                                                                              |
| .PNUMB  | = ***** GX  | 7-186      7-189      8-212      8-214      8-217      9-239      9-243      10-265      10-269    |



```

253
254
255
256
257
258
259
260
261
262
263 000154
264 000154 016700 177622
265 000160 116760 000000G 000012
266 000166 026727 000000G 000000G
267 000174 101402
268 000176 062716 000002
269 000202 000207

.SBTTL SETNLT - STORE NUMBER OF LINE TABLES
*** - SETNLT - STORE NUMBER OF LINE TABLES
INPUT:
.PNUMB = NUMBER OF LINE TABLE TO BE SET IN PROCESS SPACE
OUTPUT:
NUMBER IS VALIDATED AND THE TRANSITION IS ACCEPTED/REJECTED
:-
SETNLT
MOV TEMP,RO ; GET THE TEMPLATE ADDRESS
MGOB .PNUMB,C.NLT(RO) ; STORE THE LINE COUNT
CMP .PNUMB,#$MXNLT ; VALUE IN RANGE ?
BLOS 10$; IF LOS, YES
ADD #2,(SP) ; ELSE, REJECT
10$: RETURN

```

```

233 .SBTTL SETNUM - SET REMOTE DTE NUMBER
234 .SBTTL SETSLO - SET SUBADDRESS LOW RANGE
235 .SBTTL SETSHI - SET SUBADDRESS HIGH RANGE
236 .SBTTL SETCUG - SET USER GROUP NAME
237
238 ;+
239 ; ** SETNUM - SET REMOTE DTE NUMBER
240 ; ** SETSLO - SET SUBADDRESS LOW RANGE
241 ; ** SETSHI - SET SUBADDRESS HIGH RANGE
242 ; ** SETCUG - SET USER GROUP NAME
243 ;
244 : INPUTS .TPARS VARIABLES
245 : OUTPUT VALUE SET IN TEMPLATE.
246 :-
247
248 000314 016700 000000G SETNUM: MOV .TEMP,R0 ; POINT TO TEMPLATE
249 000320 016701 000000G MOV .PSTCN,R1 ; GET LENGTH OF PARSED STRING
250 000324 006201 ASR R1 ; LENGTH IN BCD
251 000326 010160 000000G MOV R1,D$ACLN(R0) ; SAVE LENGTH OF DTE ADDRESS
252 000332 062701 000003 ADD #3,R1 ; ROUND UP TO 4 BYTE BOUNDARY
253 000336 042701 000003 BIC #3,R1 ;
254 060342 060167 000000G ADD R1,$BYTMN ; UPDATE MINIMUM ALLOCATION
255 000346 $GTR50 #D$ANUM,#5,#CS.NUM
256
257 000370 SETSLO: $GTNUM #D$ASLO,,#9999.,#CS.SLO
258
259 000416 SETSHI: $GTNUM #D$ASHI,,#9999.,#CS.SHI
260
261 000444 SETCUG: $GTR50 #D$ACUG,#2,#D$CUG
262
263 000001 .END
264

```

TMPDSC - PARSE DESTINATION CALL MACRO V05.03b Saturday 29-Jun-85 05:32 Page 8-1

Symbol table

|                   |     |                   |                     |                   |                   |
|-------------------|-----|-------------------|---------------------|-------------------|-------------------|
| CALL 000052R      | 002 | D\$CMSK= ***** GX | SETVAL 000304R      | \$ERROR= ***** GX | \$TMPDSC 000154RG |
| CALL1 000056R     | 002 | D\$CVAL= ***** GX | \$ALPHA= 000022     | \$EXIT = 000000   | \$\$\$FLG= 177777 |
| CS.DSC= ***** GX  |     | D\$CVLN= ***** GX | \$ANY = 000020      | \$FAIL = 177777   | \$\$\$KEY= 177777 |
| CS.MSK= ***** GX  |     | ERRLEN 000065R    | \$BLANK= 000006     | \$GTR50= ***** GX | \$\$\$STA= 000000 |
| CS.VAL= ***** GX  |     | ERRMSK 000000R    | \$BYTMN= ***** GX   | \$LAMBDA= 000000  | .ERROR= ***** GX  |
| D\$CKTB 000000RG  | 003 | ERRVAL 000032R    | \$CFERR= ***** GX   | \$NUMBR= 000002   | .FLAGS= ***** GX  |
| D\$CSTB 000000RG  | 002 | FY\$SEV= 000004   | \$CLERR= ***** GX   | \$RAD50= 000016   | .PSTCN= ***** GX  |
| D\$C1 000024R     | 002 | HEXDIG 000070R    | 002 \$DIGIT= 000024 | \$STRNG= 000004   | .TEMP = ***** GX  |
| D\$CLEN= ***** GX |     | REJECT 000430R    | \$DNUMB= 000014     | \$SUBXP= 000010   | .TPARS= ***** GX  |
| D\$CMLN= ***** GX |     | SETMSK 000364R    | \$EOS = 000012      | \$TALOC= ***** GX |                   |

  

|                |     |                    |
|----------------|-----|--------------------|
| . ABS. 000000  | 000 | (RW,I,GBL,ABS,OVR) |
| 000436         | 001 | (RW,I,LCL,REL,CON) |
| \$STATE 000124 | 002 | (RW,D,LCL,REL,CON) |
| \$KTAB 000000  | 003 | (RW,D,LCL,REL,CON) |
| \$KSTR 000000  | 004 | (RW,D,LCL,REL,CON) |

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 11128 Words ( 44 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11/PLUS

Elapsed time: 00:00:24.43  
 Sy:TMPDSC.V2,[132,134]TMPDSC/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPDSC

## SYMBOL CROSS REFERENCE

CREF 04.00

| SYMBOL    | VALUE       | REFERENCES                           |
|-----------|-------------|--------------------------------------|
| CS.DST    | = ***** GX  | 7-226                                |
| CS.TSK    | = ***** GX  | 7-267                                |
| C.LEN     | = 000020    | #5-140 7-221                         |
| C.NAM     | 000006      | #5-136 7-261                         |
| C.OBJ     | 000013      | #5-138 7-265                         |
| C.PRI     | 000012      | #5-137 7-263                         |
| C.TSK     | 000014      | #5-139 7-267                         |
| C.TYP     | 000004      | #5-135 7-259                         |
| DSTKTB    | 000000 RG   | #6-161 7-230                         |
| DSTSTB    | 000000 RG   | #6-161 7-231                         |
| D\$FLNX   | = ***** GX  | 7-227                                |
| ERRNAM    | 000041 R    | #5-149                               |
| ERROBJ    | 000133 R    | #5-151                               |
| ERRPRI    | 000102 R    | #5-150                               |
| ERRTSK    | 000171 R    | #5-152                               |
| ERRTYP    | 000000 R    | #5-148                               |
| EX\$SEV   | = 000004    | #5-147 5-148 5-149 5-150 5-151 5-152 |
| SETNAM    | 000352 R    | #7-261                               |
| SETOBJ    | 000406 R    | #7-265                               |
| SETPRI    | 000366 R    | #7-263                               |
| SETTSK    | 000426 R    | #7-267                               |
| SETTYP    | 000336 R    | #7-259                               |
| \$ALPHA   | = 000022    | #6-161                               |
| \$ANY     | = 000020    | #6-161                               |
| \$BLANK   | = 000006    | #6-161                               |
| \$BYTMN   | = ***** GX  | *7-227                               |
| \$CFERR   | = ***** GX  | 7-236                                |
| \$DIGIT   | = 000024    | #6-161                               |
| \$DNUMB   | = 000014    | #6-161                               |
| \$EOS     | = 000012    | #6-161                               |
| \$ERROR   | = ***** GX  | *7-237                               |
| \$EXIT    | = 000000    | #6-161                               |
| \$FAIL    | = 177777    | #6-161                               |
| \$GPRM    | = *****     | 6-161                                |
| \$GTNMB   | = ***** GX  | 7-263                                |
| \$GTR50   | = ***** GX  | 7-259 7-261 7-267                    |
| \$LAMDA   | = 000000    | #6-161                               |
| \$NUMBR   | = 000002    | #6-161                               |
| \$RAD50   | = 000016    | #6-161                               |
| \$RONLY   | = *****     | 6-161 6-161                          |
| \$STRNG   | = 000004    | #6-161                               |
| \$SUBXP   | = 000010    | #6-161                               |
| \$TALOC   | = ***** GX  | 7-222                                |
| \$TMDST   | = 000224 RG | #7-219                               |
| \$\$\$FLG | = 177777    | #6-161                               |
| \$\$\$KEY | = 177777    | #6-161                               |
| .ERROR    | = ***** GX  | *7-228 7-234                         |
| .FLAGS    | = ***** GX  | *7-225 7-267                         |
| .TEMP     | = ***** GX  | *7-223                               |
| .TPARS    | = ***** GX  | 7-232                                |



TMPDTE - PARSE LOCAL DTE DESCR MACRO V05.03b Saturday 29-Jun-85 05:33 Page 8-2

Symbol table

|                  |                       |                     |                   |                    |
|------------------|-----------------------|---------------------|-------------------|--------------------|
| ADRSTR 000140R   | 002 DTESTB 000000RG   | 002 REJECT 000474R  | \$BYTMN= ***** GX | \$STRNG= 000004    |
| ADR1 000142R     | 002 DTE1 000106R      | 002 SETADR 000402R  | \$CFERR= ***** GX | \$SUBXP= 000010    |
| CS.DTE= ***** GX | ERRADR 000000R        | SETCON 000574R      | \$CLERR= ***** GX | \$TALOC= ***** GX  |
| CS.NET= ***** GX | ERRCT 000123R         | SETCT 000502R       | \$DIGIT= 000024   | \$TMDTE 000264RG   |
| CS.TRB= ***** GX | ERRHSH 000065R        | SETDEV 000560R      | \$DNUMB= 000014   | \$\$\$FLG= 177777  |
| CS.UNI= ***** GX | ERRLIN 000034R        | SETHSH 000416R      | \$EOS = 000012    | \$\$\$KEY= 000001  |
| C.ADR 000004     | ERRNET 000223R        | SETNET 000522R      | \$ERROR= ***** GX | \$\$\$STA= 000000  |
| C.CT 000032      | ERRSTA 000171R        | SETSTA 000544R      | \$EXIT = 000000   | \$\$\$TMP= 000003R |
| C.HSH 000030     | EX\$SEV= 000004       | SETTRB 000642R      | \$FAIL = 177777   | .ERROR= ***** GX   |
| C.LEN = 000040   | LINID 000152R         | 002 SEUNI 000614R   | \$GTNUM= ***** GX | .FLAGS= ***** GX   |
| C.LIN 000016     | LIN1 000156R          | 002 STA 000262R     | \$GTR50= ***** GX | .PNUMB= ***** GX   |
| C.NET 000034     | LIN3 000200R          | 002 \$ALPHA= 000022 | \$LAMDA= 000000   | .TEMP = ***** GX   |
| C.STA 000026     | LIN4 000214R          | 002 \$ANY = 000020  | \$NUMB= 000002    | .TPARS= ***** GX   |
| DTEKTB 000000RG  | 003 L\$LENX= ***** GX | \$BLANK= 000006     | \$RAD50= 000016   |                    |

. ABS. 000040 000 (RW,I,GBL,ABS,OVR)  
 000670 001 (RW,I,LCL,REL,CON)  
 \$STATE 000222 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000004 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000007 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 11032 Words ( 44 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:39.29  
 SY:TMPDTE.V2,[132,134]TMPDTE/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPDTE

```

234 .SBTTL SETMSK - SET UP EVENT MASKS
235
236 ;+
237 ;*** - SETMSK - SETUP EVENT MASKS
238 ;THIS ACTION ROUTINE SETS UP THE EVENT MASKS.
239
240 ;INPUT:
241 ;.PNUMB = EVENT MASK
242 ;TEMP = TEMPLATE ADDRESS
243
244 ;OUTPUT:
245 ;C.EVT = EVENT MASKS
246
247 ;-
248
249 SETMSK:
250 000240 005767 000000G TST .PNUMB ; MUST BE WORD VALUE
251 000244 001016 BNE 10$; BR IF NOT WORD VALUE
252 000246 016700 177526 MOV TEMP,R0 ; POINT AT TEMPLATE
253 000252 062700 000016 ADD #C.EVT+10,R0 ; POINT TO END OF EVENT MASKS
254 000256 020067 177520 CMP R0,STRADD ; MORE THAN 4 EVENTS?
255 000262 001407 BEQ 10$; BR IF YES - ILLEGAL
256 000264 016777 000000G 177510 MOV .PNUMB,@STRADD ; SAVE THE EVENT MASK
257 000272 062767 000002 177502 ADD #2,STRADD ; POINT TO NEXT EVENT MASK WORD
258 000300 000402 BR 20$; FINISH UP
259 000302 062716 000002 10$: ADD #2,(SP) ; REJECT THE TRANSITION
260 000306 000207 20$: RETURN

```

```

197 .SBTTL $TMSE - Setup remote characteristics template
198 ;+
199 ; *** - $TMSE - Setup remote characteristics template
200
201 ; Input:
202 ; R3-R5 - TPARS registers
203
204 ; Output:
205 ; C-bit = success/failure
206 ; If success, the remote template is stored in the end of task buffer.
207
208 ; -
209
210 $TMFIL::
211 MOV R5, -(SP) ; Save R5
212 MOV #F$LEN, R1 ; Get length of needed allocation
213 CALL $TALOC ; Try to allocate a core block
214 MOV R0, TEMP ; Save template address
215 TST (R0)+ ; Point to C.STS
216 MOV R0, .FLAGS ; Save for option setting
217 MOV #CS.FIL, (R0) ; Indicate that this is a DST$DF template
218 CLR .ERROR ; No error detail yet
219 CLR R1 ; Ignore blanks
220 MOV #FILKTBL, R2 ; Keyword table
221 MOV #FILSTBL, R5 ; State table
222 CALL TPARS ; Parse the rest of the line
223 BCC 20$; Normal return if no error
224 MOV .ERROR, R0 ; CHECK FOR ERROR DETAIL TEXT
225 BEQ 15$; IF EQ, NO DETAIL TEXT
226 ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
227 MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
228 INC $ERROR ; INDICATE SYNTAX ERROR
229 MOV (SP)+, R5 ; RESTORE R5
230 RETURN
231 RETURN

```

```

90 ;****
91 ; INITIALIZE TABLE GENERATION
92 ;****
93
94
95 000002 ISTAT$ FEASTB,FEAKTB
96
97
98 000002 STATES$
99 000002 TRANS$ $ALPHA,,SETDV1
100
101 000002 STATES$
102 000002 TRANS$ $ALPHA,,SETDV2
103
104 000002 STATES$
105 000002 TRANS$ <','>
106
107 000002 STATES$
108 000002 TRANS$ $NUMBR,,SETWD1
109
110 000002 STATES$
111 000002 TRANS$ <','>
112
113 000002 STATES$
114 000002 TRANS$ $NUMBR,,SETWD2
115
116 000002 STATES$
117 000002 TRANS$ <','>
118
119 000002 STATES$
120 000002 TRANS$ $NUMBR,,SETWD3
121
122 ; EXIT STATE
123 ;
124 000002 STATES$ EXIT
125 000002 TRANS$ $EOS,$EXIT
126
127 000002 STATE$

```

```

53 ;****
54 ; LOCAL MACROS
55 ;****
56
57 .MACRO PRINT TEXT
58 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
59 CALL $PRINT ; PRINT MESSAGE
60 .ENDM PRINT
61
62 ;****
63 ; MACRO CALLS
64 ;****
65
66 .MCALL ISTAT$,STATE$,TRANS$,MSGDF$,LLCTP$
67
68 000000 LLCTP$ LIST ; DEFINE LLC$DF TEMPLATE OFFSETS
69
70 ;
71 ; DEFINE LLC TEMPLATE
72 ;
73 .ASECT
74 .=0
75 .BLKW 1 ; C.LNK
76 .BLKW 1 ; C.STS
77 C.NAM: .BLKW 1 ; LLC PROCESS NAME
78 C.FLG: .BLKW 1 ; LLC FLAGS WORD
79 C.PRI: .BLKB 1 ; LLC PROCESS PRIORITY
80 C.LIN: .BLKB 1 ; NUMBER OF LINES ASSIGNED TO THIS LLC
81 C.DEV: .BLKW 1 ; PSEUDO DEVICE NAME FOR THIS LLC
82 C.NLT: .BLKB 1 ; NUMBER OF LINE TABLE TO ALLOCATE
83 C.EXT: .BLKB 1 ; PROCESS EXTENSION IN BLOCKS
84 C.CTM: .BLKW 1 ; COUNTER TIMER
85 CL.LEN=.
86 .PSECT
87 C.NLTD = 12 ; NLT FOR DDM/DLC PROCESSES
88 C.NCTD = 13 ; NCT FOR DDM/DLC PROCESSES
89 MSGDF$
90
91 ;****
92 ; LOCAL DATA
93 ;****
94
95 FLAG: .BLKW 1 ; FLAGS WORD
96 TEMP: .BLKW 1 ; ADDRESS OF LLC TEMPLATE

```

TMPLLC CREATED BY MACRO ON 29-JUN-85 AT 05:36 PAGE 1 B 10  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL    | VALUE      | REFERENCES                                                  |
|-----------|------------|-------------------------------------------------------------|
| CL.LEN    | = 000020   | #4-68 6-189                                                 |
| CS.LLC    | = ***** GX | 6-192                                                       |
| C.CTM     | 000016     | #4-68 *14-390                                               |
| C.DEV     | 000012     | #4-68 *11-322                                               |
| C.EXT     | 000015     | #4-68 *13-367                                               |
| C.FLG     | 000006     | #4-68 *8-246                                                |
| C.LIN     | 000011     | #4-68 *10-298                                               |
| C.NAM     | 000004     | #4-68 *7-225                                                |
| C.NCTD    | = 000013   | #4-68                                                       |
| C.NLT     | 000014     | #4-68 6-193 *12-345                                         |
| C.NLTD    | = 000012   | #4-68                                                       |
| C.PRI     | 000010     | #4-68 *9-270                                                |
| C.STS     | = ***** GX | *6-192                                                      |
| FLAG      | 000000 R   | #4-75 *6-188 8-246 8-247                                    |
| LLCKTB    | 000000 RG  | #5-82 6-197                                                 |
| LLCSTB    | 000000 RG  | #5-82 6-198                                                 |
| SETCTM    | 000406 R   | #14-388                                                     |
| SETDEV    | 000270 R   | #11-320                                                     |
| SETEXT    | 000356 R   | #13-365                                                     |
| SETFLG    | 000134 R   | #8-244                                                      |
| SETLIN    | 000222 R   | #10-294                                                     |
| SETNAM    | 000074 R   | #7-220                                                      |
| SETNLT    | 000326 R   | #12-343                                                     |
| SETPRI    | 000164 R   | #9-268                                                      |
| TEMP      | 000002 R   | #4-76 *6-191 7-224 8-245 9-269 10-295 11-321 12-344 13-366  |
| ZF.LLC    | = ***** GX | 14-389                                                      |
| \$ALPHA   | = 000022   | 8-247                                                       |
| \$ANY     | = 000020   | #5-82                                                       |
| \$BLANK   | = 000006   | #5-82                                                       |
| \$CATS    | = ***** GX | 7-223                                                       |
| \$DIGIT   | = 000024   | #5-82                                                       |
| \$DNUMB   | = 000014   | #5-82                                                       |
| \$EOS     | = 000012   | #5-82                                                       |
| \$ERROR   | = ***** GX | *6-201                                                      |
| \$EXIT    | = 000000   | #5-82                                                       |
| \$FAIL    | = 177777   | #5-82                                                       |
| \$GPRM    | = *****    | 5-82                                                        |
| \$LAMDA   | = 000000   | #5-82                                                       |
| \$MXNLT   | = ***** GX | 12-346                                                      |
| \$NLN     | = ***** GX | *10-296 *10-297                                             |
| \$NUMBR   | = 000002   | #5-82                                                       |
| \$RAD50   | = 000016   | #5-82                                                       |
| \$RONLY   | = *****    | 5-82 5-82                                                   |
| \$STRNG   | = 000004   | #5-82                                                       |
| \$SUBXP   | = 000010   | #5-82                                                       |
| \$TALOC   | = ***** GX | 6-190                                                       |
| \$TMLLC   | 000004 RG  | #6-186                                                      |
| \$\$\$FLG | = 177777   | #5-82                                                       |
| \$\$\$KEY | = 177777   | #5-82                                                       |
| .PNUMB    | = ***** GX | 9-270 9-271 9-273 10-296 10-298 10-301 12-345 12-346 13-367 |
|           |            | 13-368 14-390 14-393                                        |

TMPLOG      CREATED BY    MACRO    ON 29-JUN-85 AT 05:36      PAGE 2      B 11  
MACRO CROSS REFERENCE      CREF    C    00

MACRO NAME      REFERENCES

|         |       |        |        |        |
|---------|-------|--------|--------|--------|
| DBGTP\$ | #6-99 |        |        |        |
| ISTAT\$ | #4-69 | 6-99   |        |        |
| MTRAN\$ | #6-99 |        |        |        |
| PRINT   | #4-60 |        |        |        |
| STAT\$  | #4-69 | 6-102  | #6-108 | #6-111 |
| TRAN\$  | #4-69 | #6-103 | #6-109 |        |

Symbol table

|                  |                   |                    |                   |                        |
|------------------|-------------------|--------------------|-------------------|------------------------|
| CS.NOD= ***** GX | NODKTB 000000RG   | 003 SETNMM 000312R | \$EXIT = 000000   | \$TALOC= ***** GX      |
| C.HOS 000054     | NODSTB 000000RG   | 002 TEMP 000000R   | \$FAIL = 177777   | \$TMNOD 000002RG       |
| C.LEN = 000056   | NO\$EXA= ***** GX | \$ALPHA= 000022    | \$LAMDA= 000000   | \$\$\$FLG= 177777      |
| C.NAM 000004     | NO\$HOA= ***** GX | \$ANY = 000020     | \$MVASC= ***** GX | \$\$\$KEY= 177777      |
| C.NID 000010     | RA = 000076       | \$BLANK= 000006    | \$NTADD= ***** GX | \$\$\$STA= 000054R 002 |
| C.NUM 000052     | SETHOA 000352R    | \$CAT5 = ***** GX  | \$NTNAM= ***** GX | .PNUMB= ***** GX       |
| C.STS = ***** GX | SETHOS 000422R    | \$DIGIT= 000024    | \$NUMBR= 000002   | .PSTCN= ***** GX       |
| ID 000054R 002   | SETNAM 000100R    | \$DNUMB= 000014    | \$RAD50= 000016   | .PSTPT= ***** GX       |
| LA = 000074      | SETNID 000166R    | \$EOS = 000012     | \$STRNG= 000004   | .TPARS= ***** GX       |
| NN2 000030R 002  | SETNNA 000236R    | \$ERROR= ***** GX  | \$SUBXP= 000010   |                        |

. ABS. 000056 000 (RW,I,GBL,ABS,OVR)  
000456 001 (RW,I,LCL,REL,CON)  
\$STATE 000064 002 (RW,D,LCL,REL,CON)  
\$KTAB 000000 003 (RW,D,LCL,REL,CON)  
\$KSTR 000000 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10422 Words ( 41 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:24.31  
SY:TMPNOD.V2,[132,134]TMPNOD/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETI IB/ML,[132,10]TMPNOD



```

287 .SBTTL SETFLG - SETUP THE FLAGS BYTE
288 ;+
289 ;*** - SETFLG - SET FLAGS BYTE
290 ;
291 ;INPUT:
292 ;.PNUMB = FLAGS BYTE
293 ;.PNUMH = FLAGS BYTE (HIGH ORDER)
294 ;TEMP = TEMPLATE ADDRESS
295 ;
296 ;OUTPUT:
297 ;NUMBER IS BIT SET INTO C.FLG OF TEMPLATE
298 ;C-BIT = SUCCESS/FAILURE
299 ;
300 ;-
301
302 000322 SETFLG:
303 000322 MOV TEMP,R0 ; POINT R0 AT TEMPLATE
304 000326 016700 177452 000000G 000005 BISB .PNUMB,C.FLG(R0) ; STORE THE FLAGS BYTE
305 000334 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
306 000340 001007 BNE 10$; IF NE, NO .. REJECT
307 000342 105767 000001G TSTB .PNUMB+1 ; IS THIS A BYTE VALUE ?
308 000346 001004 BNE 10$; IF NE, NO .. BAD FLAG
309 000350 032767 000007 000000G BIT #7,.PNUMB ; LOWER 3 BITS ARE FOR ACCESS VERIFICATION
310 000356 001402 BEQ 20$; IF EQ, OKAY
311 000360 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
312 000364 000207 10$:
312 000364 000207 20$: RETURN

```

```

221 .SBTTL SETNM1 - SET UP PARTITION NAME 1
222 ;+
223 *** - SETNM1 - SETUP PARTITION NAME 1
224 ;
225 THIS ACTION ROUTINE SETS UP THE PARTITION NAME 1.
226 ;
227 INPUT:
228 .PSTPT = START ADDRESS OF THE PARTITION NAME 1 IN ASCII
229 .PSTCN = NUMBER OF CHARACTERS IN THE PARTITION NAME 1
230 TEMP = TEMPLATE ADDRESS
231 ;
232 OUTPUT:
233 NAM1 = RAD50 PARTITION NAME 1 (2 WORDS)
234 ;
235 ;
236 ;
237 SETNM1:
238 000164 MOV .PSTPT,R0 ; GET ADDRESS OF ASCII PARTITION NAME 1
239 000170 MOV PC,R1 ; PERIODS ARE ACCEPTABLE
240 000172 004767 000000G CALL $CAT5 ; CONVERT PARTITION NAME 1 TO RAD50
241 000176 016702 177600 MOV TEMP,R2 ; GET START OF TEMPLATE
242 000202 010162 000010 MOV R1,C.NAM1(R2) ; STORE PARTITION NAME 1 IN TEMPLATE
243 000206 103407 S$ BCS S$; IF CS, LESS THAN 3 CHARS WERE RAD50
244 000210 010701 MOV PC,R1 ; ELSE, PERIODS ARE ACCEPTABLE
245 000212 004767 000000G CALL $CAT5 ; CONVERT THE REST OF THE PARTITION NAME
246 000216 016702 177600 MOV TEMP,R2 ; GET START ADDRESS OF TEMPLATE
247 000222 010162 000012 MOV R1,C.NAM1+2(R2) ; STORE THE SECOND WORD
248 000226 022767 000006 000000G 5$: CMP #6,.PSTCN ; IS THE PARTITION NAME 1 TOO BIG ?
249 000234 103002 BHIS 10$; IF HIS, NO
250 000236 062716 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
251 000242 000207 10$: RETURN

```

```

154 .SBTTL TPARS STATE TABLES
155 ;
156 ; INITIALIZE TABLES
157 ;
158 ;
159 000226 ISTAT$ PSNSTB, PSNKTB
160 ;
161 ; PROCESS 'PSN$DF'
162 ;
163 000226 STATES$
164 000226 TRANS$ $LAMDA,, $CLERR,ERRNAM,.ERROR ;PSN NAME
165 000226 STATES$
166 000226 TRANS$ $STRNG,,SETNAM
167 000226 STATES$
168 000226 TRANS$ '<','>'
169 ;
170 000226 STATES$
171 000226 TRANS$ $LAMDA,, $CLERR,ERRPOR,.ERROR ;PORT NUMBER
172 000226 STATES$
173 000226 TRANS$ $NUMBR,PSN2,SETPOR
174 000226 TRANS$ $LAMDA
175 ;
176 000226 STATES$ PSN2
177 000226 TRANS$ $EOS,$EXIT
178 000226 TRANS$ '<','>'
179 ;
180 000226 STATES$
181 000226 TRANS$ $LAMDA,, $CLERR,ERRFLG,.ERROR ;OPTIONAL FLAGS
182 000226 STATES$
183 000226 TRANS$ $NUMBR,PSN3,SETFLG
184 000226 TRANS$ $LAMDA
185 ;
186 000226 STATES$ PSN3
187 000226 TRANS$ $EOS,$EXIT
188 000226 TRANS$ '<','>'
189 ;
190 000226 STATES$
191 000226 TRANS$ $LAMDA,, $CLERR,ERRTRN,.ERROR ;OPTIONAL TRAN. LOW
192 000226 STATES$
193 000226 TRANS$ $NUMBR,PSN4,SETTLO
194 000226 TRANS$ $LAMDA
195 ;
196 000226 STATES$ PSN4
197 000226 TRANS$ $EOS,$EXIT
198 000226 TRANS$ '<','>'
199 ;
200 000226 STATES$
201 000226 TRANS$ $LAMDA,, $CLERR,ERRTRN,.ERROR ;OPTIONAL TRAN. HIGH
202 000226 STATES$
203 000226 TRANS$ $NUMBR,PSN5,SETTHI
204 000226 TRANS$ $LAMDA
205 ;
206 000226 STATES$ PSN5
207 000226 TRANS$ $EOS,$EXIT
208 ;
209 ; FINAL STATE
210 ;

```

B 16

```

128 .SBTTL LOCAL DATA
129 .NLIST BEX
130 ;
131 ; SYNTAX ERROR DETAIL STRINGS
132 ;
133 .NLIST BEX
134 EX$SEV = 4
135 000000 004 103 106 ERRNAM: .ASCIZ <EX$SEV>/CFE -- Illegal PVC name/
136 000031 004 103 106 ERRLCN: .ASCIZ <EX$SEV>/CFE -- Illegal logical channel number/
137 000100 004 103 106 ERRCT: .ASCIZ <EX$SEV>/CFE -- Error in counter timer/
138 000137 004 103 106 ERRPSZ: .ASCIZ <EX$SEV>/CFE -- Illegal maximum data size/
139 000201 004 103 106 ERRWSZ: .ASCIZ <EX$SEV>/CFE -- Illegal window size/
140 000235 004 103 106 ERROWN: .ASCIZ <EX$SEV>/CFE -- Illegal owner process/
141 000273 004 103 106 ERRFLG: .ASCIZ <EX$SEV>/CFE -- Error in flags/
142 000322 004 103 106 ERRSTA: .ASCIZ <EX$SEV>/CFE -- Illegal state/
143 .EVEN
144 .LIST BEX
145
146 STA: .BLKW 1 ; STATE VALUE
147 DLM: .BLKW 1 ; DLM NAME

```

C 16

```

271 .SBTTL SETNCT - STORE NUMBER OF CONTROLLER TABLES
272 ;+
273 ; *** - SETNCT - STORE NUMBER OF CONTROLLER TABLES
274 ;
275 ; INPUT:
276 ; .PNUMB = NUMBER OF CONTROLLER TABLE TO BE SET IN PROCESS SPACE
277 ;
278 ; OUTPUT:
279 ; NUMBER IS VALIDATED AND THE TRANSITION IS ACCEPTED/REJECTED
280 ;
281 ; -
282
283 SETNCT:
284 000204 MOV TEMP,R0 ; GET THE TEMPLATE ADDRESS
285 000204 016700 177572 MOVB .PNUMB,C.NCT(R0) ; STORE THE CONTROLLER COUNT
286 000210 116760 000000G 000013 CMP .PNUMB,#$MXNCT ; VALUE IN RANGE ?
287 000216 026727 000000G 000000G BLOS 10$, ; IF LOS, YES
288 000224 101402 ADD #2,(SP) ; ELSE, REJECT
289 000226 062716 000002 10$: RETURN
290 000232 000207
291
292 .END
000001

```

Symbol table

|         |          |     |                   |          |          |                   |                   |
|---------|----------|-----|-------------------|----------|----------|-------------------|-------------------|
| ADR1    | 000130R  | 002 | D\$ACUG= ***** GX | SETCUG   | 000444R  | \$DNUMB= 000014   | \$SUBXP= 000010   |
| CS.DSA= | ***** GX |     | D\$ADLN= ***** GX | SETNUM   | 000314R  | \$EOS = 000012    | \$TALOC= ***** GX |
| CS.NUM= | ***** GX |     | D\$ALEN= ***** GX | SETSHI   | 000416R  | \$ERROR= ***** GX | \$TMDSA 000210RG  |
| CS.SHI= | ***** GX |     | D\$ANUM= ***** GX | SETSLO   | 000370F  | \$EXIT = 000000   | \$\$\$FLG= 177777 |
| CS.SLO= | ***** GX |     | D\$ASHI= ***** GX | \$ALPHA= | 000022   | \$FAIL = 177777   | \$\$\$KEY= 177777 |
| D\$CUG= | ***** GX |     | D\$ASLO= ***** GX | \$ANY =  | 000020   | \$GTNUM= ***** GX | \$\$\$STA= 000000 |
| DSAKTB  | 000000RG | 003 | ERRCLG 000157R    | \$BLANK= | 000006   | \$GTR50= ***** GX | .ERROR= ***** GX  |
| DSASTB  | 000000RG | 002 | ERRNUM 000000R    | \$BYTMN= | ***** GX | \$LAMD= 000000    | .FLAGS= ***** GX  |
| DSA1    | 000026R  | 002 | ERRSHI 000111R    | \$CFERR= | ***** GX | \$NUMBR= 000002   | .PSTCN= ***** GX  |
| DSA2    | 000052R  | 002 | ERRSLO 000044R    | \$CLERR= | ***** GX | \$RAD50= 000016   | .TEMP = ***** GX  |
| DSA3    | 000076R  | 002 | EX\$SEV= 000004   | \$DIGIT= | 000024   | \$STRNG= 000004   | .TPARS= ***** GX  |
| E       | 000122R  | 002 |                   |          |          |                   |                   |

. ABS. 000000 000 (RW,I,GBL,ABS,QVR)  
 000466 001 (RW,I,LCL,REL,CON)  
 \$STATE 000144 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000000 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000000 004 (RW,D,LCL,REL,CON)

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 11040 Words ( 44 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:28.34  
 SY:TMPDSA.V2,[132,134]TMPDSA/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPDSA

TMPDSC CREATED BY MACRO ON 29-JUN-85 AT 05:32 PAGE 1 C 3  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL    | VALUE      | REFERENCES               |
|-----------|------------|--------------------------|
| CS.DSC    | = ***** GX | 7-208                    |
| CS.MSK    | = ***** GX | 8-254                    |
| CS.VAL    | = ***** GX | 8-247                    |
| DSC.KTB   | 000000 RG  | #6-143 7-211             |
| DSC.STB   | 000000 RG  | #6-143 7-212             |
| DSC.LEN   | = ***** GX | 7-203                    |
| DSC.MLN   | = ***** GX | 7-216 *8-253             |
| DSC.MSK   | = ***** GX | 8-254                    |
| DSC.VAL   | = ***** GX | 8-247                    |
| DSC.VLN   | = ***** GX | 7-216 *8-243             |
| ERR.LEN   | 000065 R   | #5-134 7-218             |
| ERR.MSK   | 000000 R   | #5-132                   |
| ERR.VAL   | 000032 R   | #5-133                   |
| EX\$SEV   | = 000004   | #5-131 5-132 5-133 5-134 |
| REJECT    | 000430 R   | 8-241 8-251 #8-256       |
| SET.MSK   | 000364 R   | #8-249                   |
| SET.VAL   | 000304 R   | #8-239                   |
| \$ALPHA   | = 000022   | #6-143                   |
| \$ANY     | = 000020   | #6-143                   |
| \$BLANK   | = 000006   | #6-143                   |
| \$BYTMN   | = ***** GX | *8-246                   |
| \$CFERR   | = ***** GX | 7-222                    |
| \$DIGIT   | = 000024   | #6-143                   |
| \$DNUMB   | = 000014   | #6-143                   |
| \$EOS     | = 000012   | #6-143                   |
| \$ERROR   | = ***** GX | *7-223 *7-224            |
| \$EXIT    | = 000000   | #6-143                   |
| \$FAIL    | = 177777   | #6-143                   |
| \$GPRM    | = *****    | 6-143                    |
| \$GTR50   | = ***** GX | 8-247 8-254              |
| \$LAMDA   | = 000000   | #6-143                   |
| \$NUMBR   | = 000002   | #6-143                   |
| \$RAD50   | = 000016   | #6-143 6-143 6-143       |
| \$RONLY   | = *****    | 6-143                    |
| \$STRNG   | = 000004   | #6-143                   |
| \$SUBXP   | = 000010   | #6-143                   |
| \$TALOC   | = ***** GX | 7-204                    |
| \$TMPDSC  | 000154 RG  | #7-201                   |
| \$\$\$FLG | = 177777   | #6-143                   |
| \$\$\$KEY | = 177777   | #6-143                   |
| .ERROR    | = ***** GX | *7-209 *7-218 7-220      |
| .FLAGS    | = ***** GX | *7-207 8-247 8-254       |
| .PSTCN    | = ***** GX | 8-239 8-249              |
| .TEMP     | = ***** GX | *7-205 7-215 8-242 8-252 |
| .TPARS    | = ***** GX | 7-213                    |

TMPDST      CREATED BY    MACRO    ON 29-JUN-85 AT 05:33      PAGE 2      C 4

MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| CALLR      | #4-124     |        |        |        |        |        |        |        |        |        |
| DBGTP\$    | #6-161     | #6-169 | #6-190 |        |        |        |        |        |        |        |
| EPRINT     | #4-78      |        |        |        |        |        |        |        |        |        |
| ERROR\$    | #4-67      | 7-236  |        |        |        |        |        |        |        |        |
| ISTAT\$    | #4-124     | 6-161  |        |        |        |        |        |        |        |        |
| MTRANS     | #6-161     |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-62      |        |        |        |        |        |        |        |        |        |
| STAT\$     | #4-124     | 6-165  | #6-167 | #6-170 | #6-174 | #6-176 | #6-180 | #6-182 | #6-186 | #6-188 |
|            | #6-193     | #6-195 | #6-201 |        |        |        |        |        |        |        |
| TRANS      | #4-124     | #6-166 | #6-168 | #6-169 | #6-171 | #6-175 | #6-177 | #6-181 | #6-183 | #6-187 |
|            | #6-189     | #6-190 | #6-194 | #6-196 |        |        |        |        |        |        |
| \$GINMB    | #4-100     | 7-263  | 7-265  |        |        |        |        |        |        |        |
| \$GNUM     | #4-90      |        |        |        |        |        |        |        |        |        |
| \$GTR50    | #4-112     | 7-259  | 7-261  | 7-267  |        |        |        |        |        |        |



TMPDTE      CREATED BY    MACRO    ON 29-JUN-85 AT 05:33      PAGE 1      C 5  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE      | REFERENCES                                 |
|---------|------------|--------------------------------------------|
| CS.DTE  | = ***** GX | 7-265                                      |
| CS.NET  | = ***** GX | 8-317                                      |
| CS.TRB  | = ***** GX | 8-354                                      |
| CS.UNI  | = ***** GX | 8-352                                      |
| C.ADR   | 000004     | #5-135 8-298                               |
| C.CT    | 000032     | #5-140 8-315                               |
| C.HSH   | 000030     | #5-139 8-310                               |
| C.LEN   | = 000040   | #5-142 7-260                               |
| C.LIN   | 000016     | #5-136 8-348 8-350 8-352 8-354             |
| C.NET   | 000034     | #5-141 8-317                               |
| C.STA   | 000026     | #5-138 *8-330                              |
| DTEKTB  | 000000 RC  | #6-164 7-270                               |
| DTESTB  | 000000 RG  | #6-164                                     |
| ERRADR  | 000000 R   | #5-149                                     |
| ERRCT   | 000125 R   | #5-152                                     |
| ERPHSH  | 000065 R   | #5-151                                     |
| ERRLIN  | 000034 R   | #5-150                                     |
| ERRNET  | 000223 R   | #5-154                                     |
| ERRSTA  | 000171 R   | #5-153                                     |
| EX\$SEV | = 000004   | #5-148 5-149 5-150 5-151 5-152 5-153 5-154 |
| LELENX  | = ***** GX | 7-268                                      |
| REJECT  | 000474 R   | 8-304 #8-312                               |
| SETADR  | 000402 R   | #8-298                                     |
| SETCON  | 000574 R   | #8-350                                     |
| SETCT   | 000502 R   | #8-315                                     |
| SETDEV  | 000560 R   | #8-348                                     |
| SETHSH  | 000416 R   | #8-300                                     |
| SETNET  | 000522 R   | #8-317                                     |
| SETSTA  | 000544 R   | #8-329                                     |
| SETTRB  | 000642 R   | #8-354                                     |
| SETUNI  | 000614 R   | #8-352                                     |
| STA     | 000262 R   | #5-157 *7-267 8-330                        |
| \$ALPHA | = 000022   | #6-164                                     |
| \$ANY   | = 000020   | #6-164                                     |
| \$BLANK | = 000006   | #6-164                                     |
| \$BYTMN | = ***** GX | *7-268 *8-309                              |
| \$CFERR | = ***** GX | 7-276                                      |
| \$DIGIT | = 000024   | #6-164                                     |
| \$DNUMB | = 000014   | #6-164                                     |
| \$EOS   | = 000012   | #6-164                                     |
| \$ERROR | = ***** GX | *7-277 *7-278                              |
| \$EXIT  | = 000000   | #6-164                                     |
| \$FAIL  | = 177777   | #6-164                                     |
| \$GPRM  | = *****    | 6-164                                      |
| \$GTNUM | = ***** GX | 8-310 8-315 8-350 8-352 8-354              |
| \$GTR50 | = ***** GX | 8-298 8-317 8-348                          |
| \$LAMPD | = 000000   | #6-164                                     |
| \$NUMBR | = 000002   | #6-164                                     |
| \$RAD50 | = 000016   | #6-164                                     |
| \$RONLY | = *****    | 6-164 6-164                                |
| \$STRNG | = 000004   | #6-164                                     |
| \$SUBXP | = 000010   | #6-164                                     |

```

262 .SBTTL SETMON - SETUP MONITOR LOGGING SINK TYPE
263 .SBTTL SETCON - SETUP CONSOLE LOGGING SINK TYPE
264 .SBTTL SETFIL - SETUP FILE LOGGING SINK TYPE
265
266 ;*
267 ;*** - SETMON - SETUP MONITOR LOGGING SINK TYPE
268 ;*** - SETCON - SETUP CONSOLE LOGGING SINK TYPE
269 ;*** - SETFIL - SETUP FILE LOGGING SINK TYPE
270
271 ; THESE ACTION ROUTINES SET UP THE LOGGING SINK TYPE
272
273 INPUT:
274 TEMP = TEMPLATE ADDRESS
275
276 OUTPUT:
277 C.SNK = LOGGING SINK TYPE
278 C-BIT = SUCCESS/FAILURE
279
280 ; -
281
282 .ENABL LSB
283
284 000310 012701 000001 SETCON: MOV #FF.CON,R1 ; GET CONSOLE LOGGING SINK TYPE
285 000314 000405 BR 10$
286
287 000316 012701 000004 SETMON: MOV #FF.MON,R1 ; GET MONITOR LOGGING SINK TYPE
288 000322 000402 BR 10$
289
290 000324 012701 000002 SETFIL: MOV #FF.FIL,R1 ; GET FILE LOGGING SINK TYPE
291
292 10$: MOV TEMP,R0 ; POINT AT TEMPLATE
293 000330 016700 177444 MOV R1,C.SNK(R0) ; STORE LOGGING SINK TYPE
294 000334 010160 000016 RETURN
295 000340 000207
296
297 .DSABL LSB
298
299 000001 .END

```

```

233 .SBTTL Action routines
234
235 ;+
236 ;** CLEFIL - clear file identifier
237 ;
238 ; Inputs: none.
239 ;
240 ; Output: file is cleared.
241 ;-
242 CLEFIL:
243 CLR FILE ; zero offset
244 MOV #FD,R0 ; point to prototype FD
245 MOV #F.LEN,R1 ; get length
246 CLRB (R0)+ ; zero prototype
247 SOB R1,10$; ...
248 RETURN
249
250 ;+
251 ;** CKFSP - check parsed filespec
252 ;
253 ; Inputs: .TPARS variables.
254 ;
255 ; Output: FD+F.DEV= parsed filespec string
256 ; The transition is accepted/rejected
257 ;-
258 CKFSP:
259 CMP .PSTCN,#MXFSP ; Check length
260 BHI REJ ; If H1, too long
261 MOV #FD+F.DEV-1,R2 ; Else point to parsed data area
262 CALLR MOVSTR ; Move it to parsed data area
263
264 MOV #ERRFSP,.ERROR ; ELSE ERROR
265 ADD #2,(SP) ; Reject the transition
266 RETURN
267
268 ;+
269 ;** CKUIC - check parsed UIC length
270 ;
271 ; Inputs: .TPARS variables.
272 ; R3 = byte count of remainder of input string
273 ; R4 = address of remainder of input string
274 ;
275 ; Output: The transition is accepted/rejected
276 ; R0-R1 clobbered
277 ;-
278 CKUIC:
279 CMP .PSTCN,#11. ; Check length
280 BHI 10$; If H1, too long
281 BR 20$; Else, OK
282 MOV #ERRUIC,.ERROR ; ELSE ERROR
283 ADD #2,(SP) ; Reject the transition
284 RETURN
285
286 ;+
287 ;** STFIL - store parsed file spec
288 ;
289 ; Inputs: FD = prototype file descriptor
290 ; FILE = offset to address of FD in template

```

```

129 .SBTTL $TMFEA - SETUP FEATURES TEMPLATE
130
131 +
132 *** - $TMFEA - SETUP FEATURES TEMPLATE
133
134 INPUT:
135 R3-R5 - TPARS REGISTERS
136
137 OUTPUT:
138 C-BIT = SUCCESS/FAILURE
139 IF SUCCESS, THE FEATURES TEMPLATE IS STURED IN THE END OF TASK BUFFER.
140
141 -
142
143 $TMFEA::
144 000002 010546 MOV R5,-(SP) ; SAVE R5
145 000004 012701 000014 MOV #C.LEN,R1 ; GET LENGTH OF ALLOCATION
146 000010 004767 000000G CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
147 000014 010067 177760 MOV R0,TEMP ; STORE TEMPLATE ADDRESS
148 000020 012760 000000G 000000G MOV #CS.FEA,C.STS(R0) ; INDICATE THAT THIS IS A FEASDF TEMPLATE
149 000026 005001 CLR R1 ; IGNORE BLANKS
150 000030 012702 000000' MOV #FEAKTB,R2 ; KEYWORD TABLE
151 000034 012705 000000' MOV #FEASTB,R5 ; STATE TABLE
152 000040 004767 000000G CALL .TPARS ; PARSE THE REST OF THE LINE
153 000044 103002 BCC 20$; NORMAL RETURN IF NO ERROR
154 000046 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
155 000052 012605 20$: MOV (SP)+,R5 ; RESTORE R5
156 000054 000207 RETURN

```

```

78 ;****
79 ; INITIALIZE TABLE GENERATION
80 ;****
81
82 ISTAT$ LLCSTB,LLCKTB
83
84 STATES$; LLC NAME
85 TRANS$ $RAD50,,SETNAM
86
87 STATES$; FLAGS
88 TRANS$ '<','>'
89
90 STATES$
91 TRANS$ $LAMDA,BITO
92
93 STATES$ BITS
94 TRANS$ '! ,BITO
95 TRANS$ $LAMDA,PRIOR,SETFLG
96
97 STATES$ BITO
98 TRANS$!ZF
99
100 STATES$
101 TRANS$ 'LLC',BITS,,ZF.LLC,FLAG
102 TRANS$ 'TIM',BITS,,ZF.TIM,FLAG
103 TRANS$ 'MFL',BITS,,ZF.MFL,FLAG
104 TRANS$ 'MTM',BITS,,ZF.MTM,FLAG
105 TRANS$ 'COU',BITS,,ZF.COU,FLAG
106 TRANS$ 'X3P',BITS,,ZF.X3P,FLAG
107 TRANS$ 'SLI',BITS,,ZF.SLI,FLAG
108 TRANS$ 'INI',BITS,,ZF.INI,FLAG
109
110 STATES$ PRIOR ; PRIORITY
111 TRANS$ '<','>'
112
113 STATES$
114 TRANS$ $NUMBR,,SETPRI
115
116 STATES$; CHANNELS
117 TRANS$ '<','>'
118
119 STATES$
120 TRANS$ $NUMBR,,SETLIN
121
122 STATES$; DEVICE ID
123 TRANS$ $EOS,$EXIT
124 TRANS$ '<','>'
125
126 STATES$
127 TRANS$ $STRNG,NLT,SETDEV
128 TRANS$ $LAMDA
129
130 STATES$ NLT ; LINE TABLES
131 TRANS$ $EOS,$EXIT
132 TRANS$ '<','>'
133
134 STATE$

```

TMPLLC      CREATED BY    MACRO    ON 29-JUN-85 AT 05:36      PAGE 2      C 10  
SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL | VALUE            | REFERENCES                |
|--------|------------------|---------------------------|
| .PNUMH | =    *****    GX | 10-299    14-391          |
| .PSTCN | =    *****    GX | 7-226    11-324           |
| .PSTPT | =    *****    GX | 7-221    11-322    11-323 |
| .TPARS | =    *****    GX | 6-199                     |

|           |      |      |         |    |      |        |          |
|-----------|------|------|---------|----|------|--------|----------|
| TTTTTTTTT | MM   | MM   | PPPPPPP | NN | NN   | 000000 | DDDDDDDD |
| TTTTTTTTT | MM   | MM   | PPPPPPP | NN | NN   | 000000 | DDDDDDDD |
| TT        | MMMM | MMMM | PP      | PP | NN   | 00     | DD       |
| TT        | MMMM | MMMM | PP      | PP | NN   | 00     | DD       |
| TT        | MM   | MM   | PP      | PP | NNNN | 00     | DD       |
| TT        | MM   | MM   | PP      | PP | NNNN | 00     | DD       |
| TT        | MM   | MM   | PPPPPPP | NN | NN   | 00     | DD       |
| TT        | MM   | MM   | PPPPPPP | NN | NN   | 00     | DD       |
| TT        | MM   | MM   | PP      | NN | NNNN | 00     | DD       |
| TT        | MM   | MM   | PP      | NN | NNNN | 00     | DD       |
| TT        | MM   | MM   | PP      | NN | NN   | 00     | DD       |
| TT        | MM   | MM   | PP      | NN | NN   | 00     | DD       |
| TT        | MM   | MM   | PP      | NN | NN   | 00     | DD       |
| TT        | MM   | MM   | PP      | NN | NN   | 000000 | DDDDDDDD |
| TT        | MM   | MM   | PP      | NN | NN   | 000000 | DDDDDDDD |

....  
....  
....  
....

|            |          |           |
|------------|----------|-----------|
| LL         | SSSSSSSS | TTTTTTTTT |
| LL         | SSSSSSSS | TTTTTTTTT |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SSSSSS   | TT        |
| LL         | SSSSSS   | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LLLLLLLLLL | SSSSSSSS | TT        |
| LLLLLLLLLL | SSSSSSSS | TT        |

TMPNOD CREATED BY MACRO ON 29-JUN-85 AT 05:37 PAGE 1 C 12  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL   | VALUE       | REFERENCES                                                |
|----------|-------------|-----------------------------------------------------------|
| CS.NOD   | = ***** GX  | 6-171                                                     |
| C.HOS    | = 000054    | *4-89 *6-170 *11-319 *12-343                              |
| C.LEN    | = 000056    | *4-90 6-165                                               |
| C.NAM    | = 000004    | *4-84 *7-203 *7-208                                       |
| C.NID    | = 000010    | *4-86 8-233                                               |
| C.NUM    | = 000052    | *4-88 *6-169 *9-266 *10-292                               |
| C.STS    | = ***** GX  | *6-171 *9-268 *11-320                                     |
| LA       | = 000074    | *4-74                                                     |
| NODKTB   | = 000000 RG | *5-99 6-173                                               |
| NODSTB   | = 000000 RG | *5-99 6-174                                               |
| JOSEA    | = ***** GX  | 9-268                                                     |
| JOA      | = ***** GX  | 11-320                                                    |
| RA       | = 000076    | *4-75                                                     |
| SETHOA   | = 000352 R  | *11-310                                                   |
| SETHOS   | = 000422 R  | *12-337                                                   |
| SETNAM   | = 000100 R  | *7-198                                                    |
| SETNID   | = 000166 R  | *8-231                                                    |
| SETNNA   | = 000236 R  | *9-257                                                    |
| SETNNM   | = 000312 R  | *10-286                                                   |
| TEMP     | = 000000 R  | *4-76 *6-167 7-202 7-207 8-232 9-258 10-287 11-311 12-338 |
| \$ALPHA  | = 000022    | *5-99                                                     |
| \$ANY    | = 000020    | *5-99                                                     |
| \$BLANK  | = 000006    | *5-99                                                     |
| \$CAT5   | = ***** GX  | 7-201 7-206                                               |
| \$DIGIT  | = 000024    | *5-99                                                     |
| \$DNUMB  | = 000014    | *5-99                                                     |
| \$EOS    | = 000012    | *5-99                                                     |
| \$ERROR  | = ***** GX  | *6-177                                                    |
| \$EXIT   | = 000000    | *5-99                                                     |
| \$FAIL   | = 177777    | *5-99                                                     |
| \$GPRM   | = *****     | 5-99                                                      |
| \$LAMBDA | = 000000    | *5-99                                                     |
| \$MVASC  | = ***** GX  | 8-239                                                     |
| \$NTADD  | = ***** GX  | *6-168 *9-267 *10-293                                     |
| \$NTNAM  | = ***** GX  | *7-204 *7-209                                             |
| \$NUMBR  | = 000002    | *5-99                                                     |
| \$RAD50  | = 000016    | *5-99                                                     |
| \$RONLY  | = *****     | 5-99 5-99                                                 |
| \$STRNG  | = 000004    | *5-99                                                     |
| \$SUBXP  | = 000010    | *5-99                                                     |
| \$TALOC  | = ***** GX  | 6-166                                                     |
| \$TMNOD  | = 000002 RG | *6-163                                                    |
| \$%FLG   | = 177777    | *5-99                                                     |
| \$%KEY   | = 177777    | *5-99                                                     |
| .PNUMB   | = ***** GX  | 9-259 10-288 11-312 12-339                                |
| .PSTCN   | = ***** GX  | 7-210 8-235 8-240                                         |
| .PSTPT   | = ***** GX  | 7-199 8-234                                               |
| .TPARS   | = ***** GX  | 6-175                                                     |



```
314 .SBTTL SETCPY - SETUP THE MAXIMUM NUMBER OF COPIES ALLOWED
315
316 *** - SETCPY - SETUP THE MAXIMUM NUMBER OF COPIES ALLOWED
317
318 INPUT:
319 .PNUMB = THE MAX. # OF COPIES ALLOWED
320 .PNUMH = THE MAX. # OF COPIES ALLOWED (HIGH BYTE)
321 TEMP = TEMPLATE ADDRESS
322
323 OUTPUT:
324 THE NUMBER OF COPIES IS MOVED INTO C.CPY OF THE TEMPLATE
325 C-BIT = SUCCESS/FAILURE
326
327
328
329 SETCPY:
330 000366 016700 177406 MOV TEMP,R0 ; POINT AT TEMPLATE
331 000372 116760 000000G 000012 MOVB .PNUMB,C.CPY(R0) ; STORE THE NUMBER OF ALLOWED COPIES
332 000400 005767 000000G TST .PNUMH ; IS THIS A WORD VALUE ?
333 000404 001004 BNE 10$; IF NE, NO .. REJECT
334 000406 022767 000077 000000G CMP #63,..PNUMB ; IS THE NUMBER IN RANGE
335 000414 103002 BHIS 20$; IF HIS, YES .. OKAY
336 000416 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT
337 000422 000207 20$: RETURN
338
339
340 000001 .END
```

SETEXT - CHECK THE EXTENSION

```

253 .SBTTL SETEXT - CHECK THE EXTENSION
254
255 *** - SETEXT - CHECK THE POOL EXTENSION
256
257 INPUT:
258 .PNUMB = EXTENSION
259 .PNUMH = EXTENSION (HIGH ORDER)
260 TEMP = TEMPLATE ADDRESS
261
262 OUTPUT:
263 THE ACTUAL EXTENSION IS CALCULATED FROM BUFFER
264 AND NODE COUNTER ALLOCATIONS AND STORES IN $BLKXT.
265 THIS ROUTINE ACTS AS A SANITY TEST ON THE VALUE IN
266 THE CETAB.
267
268
269 000244 SETEXT: MOV TEMP,RO ; GET START OF TEMPLATE
270 000244 016700 177532 MOV .PNUMB,C.EXT(RO) ; STORE THE EXTENSION IN THE TEMPLATE
271 000250 016760 000000G 000014 TST .PNUMH ; IS IT A WORD VALUE ?
272 000256 005767 000000G BEQ 20$; IF EQ, YES
273 000262 001402 000000G 10$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
274 000264 062716 000002 20$: RETURN
275 000270 000207

```

SETAIL - SET THE BYTE-AREA EXTENSION

TMPPSN - PARSE PSN IN CETAB  
TPARS STATE TABLES

MACRO V05.03b Saturday 29-Jun-85 05:38 Page 6-1

C.15

211 000226  
212

STATE\$

TMPPSN - PARSE PSN IN CETAB  
\$TMPPSN - BUILD PSN TEMPLATE

MACRO V05.03b Saturday 29-Jun-85 05:38 Page 7

D.15

C 16

```

149 .SBTTL TPARS STATE TABLES
150 ;
151 ; INITIALIZE TABLES
152 ;
153
154 000354 ISTAT$ PVCSTB, PVCKTB
155 ;
156 ; PROCESS "PVC$DF"
157 ;
158 000354 STATES$; PVC NAME
159 000354 TRANS$ $LAMDA,, $CLERR,ERRNAM,,ERROR
160 000354 STATES$
161 000354 TRANS$!DLMPVC,PVCO,,1,DLM ;ALTERNATE FORMAT FOR DLM
162 000354 TRANS$ $STRNG,,SETNAM
163 000354 STATES$ PVCO
164 000354 TRANS$ <'>
165
166 000354 STATES$; LOGICAL CHANNEL NUMBER
167 000354 TRANS$ $LAMDA,, $CLERR,ERRLCN,,ERROR
168 000354 STATES$
169 000354 TRANS$ $NUMBR,,SETLCN
170 000354 STATES$
171 000354 TRANS$ <'>
172
173 000354 STATES$; COUNTER TIMER
174 000354 TRANS$ $LAMDA,, $CLERR,ERRCT,,ERROR
175 000354 STATES$
176 000354 TRANS$ $NUMBR,,SETCT
177 000354 STATES$
178 000354 TRANS$ <'>
179
180 000354 STATES$; STATE
181 000354 TRANS$ $LAMDA,, $CLERR,ERRSTA,,ERROR
182 000354 STATES$
183 000354 TRANS$ %ON%,PVC1,,1,STA
184 000354 TRANS$ %OFF%,PVC1
185 000354 STATES$ PVC1
186 000354 TRANS$ $EOS,$EXIT,SETSTA
187 000354 TRANS$ <'>,,SETSTA
188
189 000354 STATES$; OPTIONAL MAXIMUM BLOCK SIZE
190 000354 TRANS$ $LAMDA,, $CLERR,ERRPSZ,,ERROR
191 000354 STATES$
192 000354 TRANS$ $NUMBR,PVC2,SETPSZ
193 000354 TRANS$ $LAMDA
194
195 000354 STATES$ PVC2
196 000354 TRANS$ $EOS,$EXIT
197 000354 TRANS$ <'>
198
199 000354 STATES$; OPTIONAL WINDOW SIZE
200 000354 TRANS$ $LAMDA,, $CLERR,ERRWS?,.ERROR
201 000354 STATES$
202 000354 TRANS$ $NUMBR,PVC3,SETWSZ
203 000354 TRANS$ $LAMDA
204
205 000354 STATES$ PVC3

```

D 16

TMPDLC - PARSE DLC PROCESS DEFINI MACRO V05.03b Saturday 29-Jun-85 05:31 Page 11-1  
Symbol table

|                  |     |        |          |     |                   |                   |                    |
|------------------|-----|--------|----------|-----|-------------------|-------------------|--------------------|
| BITS 000020R     | 002 | DLCSTB | 000000RG | 002 | ZF.COU= ***** GX  | \$EOS = 000012    | \$TALOC= ***** GX  |
| BIT0 000032R     | 002 | FLAG   | 000000R  |     | ZF.DLC= ***** GX  | \$ERROR= ***** GX | \$TMDLC 000004RG   |
| CS.DLC= ***** GX |     | NCT    | 000122R  | 002 | ZF.MAN= ***** GX  | \$EXIT = 000000   | \$\$\$FLG= 177777  |
| C.FLG 000006     |     | PRIOR  | 000076R  | 002 | ZF.MFL= ***** GX  | \$FAIL = 177777   | \$\$\$KEY= 000005  |
| C.LEN = 000014   |     | SETFLG | 000124R  |     | ZF.TIM= ***** GX  | \$LAMD= 000000    | \$\$\$STA= 000000  |
| C.NAM 000004     |     | SETNAM | 000064R  |     | \$ALPHA= 000022   | \$MXNCT= ***** GX | \$\$\$TMP= 000024R |
| C.NCT 000013     |     | SETNCT | 000204R  |     | \$ANY = 000020    | \$MXNLT= ***** GX | .PNUMB= ***** GX   |
| C.NLT 000012     |     | SETNLT | 000134R  |     | \$BLANK= 000006   | \$NUMB= 000002    | .PSTCN= ***** GX   |
| C.PRI 000010     |     | SETPRI | 000140R  |     | \$CAT5 = ***** GX | \$RAD50= 000016   | .PSTPT= ***** GX   |
| C.STS = ***** GX |     | TEMP   | 000002R  |     | \$DIGIT= 000024   | \$STRNG= 000004   | .TPARS= ***** GX   |
| DLCKTB 000000RG  | 003 | ZF     | 000136R  | 002 | \$DNUMB= 000014   | \$SUBXP= 000010   |                    |

. ABS. 000014 000 (RW,I,GBL,ABS,OVR)  
000234 001 (RW,I,LCL,REL,CON)  
\$STATE 000144 002 (RW,D,LCL,REL,CON)  
\$KTAB 000014 003 (RW,D,LCL,REL,CON)  
\$KSTR 000027 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10438 Words ( 41 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:28.37  
SY:TMPDLC.V2,[132,134]TMPDLC/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPDLC

TMPDSA CREATED BY MACRO ON 29-JUN-85 AT 05:32 PAGE 1 D 2  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL    | VALUE      | REFERENCES                     |
|-----------|------------|--------------------------------|
| CS.DSA    | = ***** GX | 7-218                          |
| CS.NUM    | = ***** GX | 8-255                          |
| CS.SHI    | = ***** GX | 8-259                          |
| CS.SLO    | = ***** GX | 8-257                          |
| DSACUG    | = ***** GX | 8-261                          |
| DSAKTB    | 000000 RG  | #6-145 7-221                   |
| DSASTB    | 000000 RG  | #6-145 7-222                   |
| DSACUG    | = ***** GX | 8-261                          |
| DSADLN    | = ***** GX | *8-251                         |
| DSALEN    | = ***** GX | 7-213                          |
| DSANUM    | = ***** GX | 8-255                          |
| DSASHI    | = ***** GX | 8-259                          |
| DSASLO    | = ***** GX | 8-257                          |
| ERRCUG    | 000157 R   | #5-136                         |
| ERRNUM    | 000000 R   | #5-133                         |
| ERRSHI    | 000111 R   | #5-135                         |
| ERRSLO    | 000044 R   | #5-134                         |
| EX\$SEV   | = 000004   | #5-132 5-133 5-134 5-135 5-136 |
| SETCUG    | 000444 R   | #8-261                         |
| SETNUM    | 000314 R   | #8-248                         |
| SETSHI    | 000416 R   | #8-259                         |
| SETSLO    | 000370 R   | #8-257                         |
| \$ALPHA   | = 000022   | #6-145                         |
| \$ANY     | = 000020   | #6-145                         |
| \$BLANK   | = 000006   | #6-145                         |
| \$BYTMN   | = ***** GX | *8-254                         |
| \$CFERR   | = ***** GX | 7-227                          |
| \$DIGIT   | = 000024   | #6-145                         |
| \$DNUMB   | = 000014   | #6-145                         |
| \$EOS     | = 000012   | #6-145                         |
| \$ERROR   | = ***** GX | *7-228 *7-229                  |
| \$EXIT    | = 000000   | #6-145                         |
| \$FAIL    | = 177777   | #6-145                         |
| \$GPRM    | = *****    | 6-145                          |
| \$GTNUM   | = ***** EX | 8-257 8-259                    |
| \$GTR50   | = ***** GX | 8-255 8-261                    |
| \$LAMDA   | = 000000   | #6-145                         |
| \$NUMBR   | = 000002   | #6-145                         |
| \$RAD50   | = 000016   | #6-145                         |
| \$ROJLY   | = *****    | 6-145 6-145                    |
| \$STRNG   | = 000004   | #6-145                         |
| \$SUBXP   | = 000010   | #6-145                         |
| \$TALOC   | = ***** GX | 7-214                          |
| \$TMDSA   | 000210 RG  | #7-211                         |
| \$\$\$FLG | = 177777   | #6-145                         |
| \$\$\$KEY | = 177777   | #6-145                         |
| .ERROR    | = ***** GX | *7-219 7-225                   |
| .FLAGS    | = ***** GX | *7-217 8-255 8-257 8-259 8-261 |
| .PSTCN    | = ***** GX | 8-249                          |
| .TEMP     | = ***** GX | *7-215 8-248                   |
| .TPARS    | = ***** GX | 7-223                          |

TMPDSC      CREATED BY    MACRO    ON 29-JUN-85 AT 05:32      PAGE 2      D 3  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| CALLR      | #4-124     |        |        |        |        |        |        |        |        |        |
| DBGTP\$    | #6-143     | #6-152 | #6-160 | #6-170 | #6-174 | #6-175 | #6-176 | #6-177 | #6-178 | #6-179 |
| EPRINT     | #4-78      |        |        |        |        |        |        |        |        |        |
| ERROR\$    | #4-57      | 7-222  |        |        |        |        |        |        |        |        |
| ISTAT\$    | #4-124     | 6-143  |        |        |        |        |        |        |        |        |
| MTRANS     | #6-143     |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-62      |        |        |        |        |        |        |        |        |        |
| STATE\$    | #4-124     | 6-148  | #6-150 | #6-153 | #6-156 | #6-158 | #6-161 | #6-166 | #6-168 | #6-172 |
|            | #6-183     |        |        |        |        |        |        |        |        |        |
| TRAN\$     | #4-124     | #6-149 | #6-151 | #6-152 | #6-154 | #6-157 | #6-159 | #6-160 | #6-162 | #6-167 |
|            | #6-169     | #6-170 | #6-173 | #6-174 | #6-175 | #6-176 | #6-177 | #6-178 | #6-179 |        |
| \$GTNMB    | #4-100     |        |        |        |        |        |        |        |        |        |
| \$GTNUM    | #4-90      |        |        |        |        |        |        |        |        |        |
| \$GTR50    | #4-112     | 8-247  | 8-254  |        |        |        |        |        |        |        |

|            |      |      |          |          |            |            |      |
|------------|------|------|----------|----------|------------|------------|------|
| TTTTTTTTTT | MM   | MM   | PPPPPPPP | DDDDDDDD | TTTTTTTTTT | EEEEEEEEEE |      |
| TTTTTTTTTT | MM   | MM   | PPPPPPPP | DDDDDDDD | TTTTTTTTTT | EEEEEEEEEE |      |
| TT         | MMMM | MMMM | PP       | DD       | TT         | EE         |      |
| TT         | MMMM | MMMM | PP       | DD       | TT         | EE         |      |
| TT         | MM   | MM   | PP       | DD       | TT         | EE         |      |
| TT         | MM   | MM   | PP       | DD       | TT         | EE         |      |
| TT         | MM   | MM   | PPPPPPPP | DD       | TT         | EEEEEEEE   |      |
| TT         | MM   | MM   | PPPPPPPP | DD       | TT         | EEEEEEEE   |      |
| TT         | MM   | MM   | PP       | DD       | TT         | EE         |      |
| TT         | MM   | MM   | PP       | DD       | TT         | EE         |      |
| TT         | MM   | MM   | PP       | DD       | TT         | EE         | .... |
| TT         | MM   | MM   | PP       | DD       | TT         | EE         | .... |
| TT         | MM   | MM   | PP       | DDDDDDDD | TT         | EEEEEEEEEE | .... |
| TT         | MM   | MM   | PP       | DDDDDDDD | TT         | EEEEEEEEEE | .... |

|            |          |            |
|------------|----------|------------|
| LL         | SSSSSSSS | TTTTTTTTTT |
| LL         | SSSSSSSS | TTTTTTTTTT |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SSSSSS   | TT         |
| LL         | SSSSSS   | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LLLLLLLLLL | SSSSSSSS | TT         |
| LLLLLLLLLL | SSSSSSSS | TT         |



TMPDTE      CREATED BY    MACRO    ON 29-JUN-85 AT 05:33      PAGE 2      D 5

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE       | REFERENCES                              |
|-----------|-------------|-----------------------------------------|
| \$TALOC   | = ***** GX  | 7-261                                   |
| \$TMDTE   | = 000264 RG | #7-258                                  |
| \$\$\$FLG | = 177777    | #6-164                                  |
| \$\$\$KEY | = 177777    | #6-164                                  |
| .ERROR    | = ***** GX  | *7-266      7-274                       |
| .FLAGS    | = ***** GX  | *7-264      8-317      8-352      8-354 |
| .PNUMB    | = ***** GX  | 8-302                                   |
| .TEMP     | = ***** GX  | *7-262      8-329                       |
| .TPARS    | = ***** GX  | 7-272                                   |

```

Symbol table
CS.EVT= ***** GX FF.LIN= 000010 F.LENX= ***** GX STRADD 000002R $QINSN= ***** GX
C.CLS 000004 FF.MOD= 000100 F.LIN 000020 TEMP 000000R $RAD50= 000016
C.EVT 000006 FF.MON= 000004 F.LNK 000000 $ALPHA= 000022 $STRNG= 000004
C.LEN = 000020 FF.MSK= 000077 F.MOD 000022 $ANY = 000020 $SUBXP= 000010
C.SNK 000016 FF.PRT= 000200 F.REM 000026 $BLANK= 000006 $TMEVT 000004RG
C.STS = ***** GX FF.QUL= 000370 F.SEV 000004 $BYTMN= ***** GX $TMLST= ***** GX
EVTKT B 000000RG 003 FF.REM= 100000 LBRA = 000074 $DIGIT= 000024 $TTALC= ***** GX
EVTMSK 000042R 002 FINEVT 000122R MAXCLS= 077700 $DNUMB= 000014 $$$FLG= 177777
EVTSTB 000000RG 002 F.ADD 000016 RBRA = 000076 $EOS = 000012 $$$KEY= 00 002
EXIT 000036R 002 F.CEV 000030 SETCLS 000166R $ERROR= ***** GX $$$STA= 000042R 002
FF.ADD= 000020 F.CIR 000024 SETCON 000310R $EXIT = 000000 $$$TMP= 000015R 004
FF.CIR= 000040 F.CLS 000002 SETFIL 000324R $FAIL = 177777 .PNUMB= ***** GX
FF.CON= 000001 F.EVT 000004 SETMON 000316R $LAMBDA= 000000 .PNUMH= ***** GX
FF.FIL= 000002 F.FLG 000014 SETMSK 000240R $NUMBR= 000002 .TPARS= ***** GX
FF.HST= 040000 F.LEN 000016
. ABS. 000040 000 (RW,I,GBL,ABS,OVR)
 000342 001 (RW,I,LCL,REL,CON)
$STATE 000056 002 (RW,D,LCL,REL,CON)
$KTAB 000006 003 (RW,I,LCL,REL,CON)
$KSTR 000025 004 (RW,I,LCL,REL,CON)
Errors detected: 0

```

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 11160 Words ( 44 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:20.52  
SY:TMPEVT.V2,[132,134]TMPEVT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPEVT

```

290 ; $FILHD = single word FD listhead
291 ; $BYTMN = minimum bytes in byte allocated pool
292 ; .TEMP = address of FIL$DF template
293
294 ; Output: If new file, an FD is linked into $FILHD
295 ; and $BYTMN is updated - otherwise use count
296 ; of existing FD is incremented.
297 ;
298
299 000336 004567 000000G STFIL: JSR R5,$$AVRG ; Save R3-R5
300 000342 012700 000000G MOV #$FILHD,R0 ; Point to file list
301
302 ; Look for parsed file names in file descriptor list
303
304 000346 010005 10$: MOV R0,R5 ; Potential link address
305 000350 011000 MOV (R0),R0 ; Next file block
306 000352 001415 BEQ 30$, ; If EQ, not found - allocate one
307 000354 010001 MOV R0,R1 ; copy file block pointer
308 000356 062701 000004 ADD #F.DEV-1,R1 ; skip don't care info
309 000362 012702 000006 MOV #FD+F.DEV-1,R2 ; point to prototype FD
310 000366 012703 000037 MOV #F.LEN-<F.DEV-1>,R3 ; length to compare
311
312 000372 122122 20$: CMPB (R1)+,(R2)+ ; Compare a byte
313 000374 001364 BNE 10$, ; If NE, no match
314 000376 077303 SOB R3,20$; Countdown
315
316 ; Already parsed: just up the use count
317
318 000400 005260 000002 INC F.CNT(R0) ; It matched - increase the use count
319 000404 000427 BR 40$; And point template to it
320
321 ; Allocate a new file descriptor and fill it in
322
323 000406 012701 30$: MOV #F.LEN,R1 ; Get length of needed allocation
324 000412 004767 CALL $T1ALC ; Try to allocate a core block
325 000416 010015 MOV R0,(R5) ; Link it at the end
326 000420 010003 MOV R0,R3 ; Copy pointer to new FD
327 000422 005023 CLR (R3)+ ; Clear link address
328 000424 012723 000001 MOV #1,(R3)+ ; Initialize use count
329 000430 012701 000037 MOV #F.LEN-<F.DEV-1>,R1 ; Maximum parsed file size
330 000434 012702 000006 MOV #FD+F.DEV-1,R2 ; Point to parsed file in proto FD
331
332 000440 112223 35$: MOVB (R2)+,(R3)+ ; MOVE...
333 000442 077102 SOB R1,35$; ...IT
334
335 ; Calculate addition to byte allocated pool
336
337 000444 116701 MOVB FD+F.DEV-1,R1 ; Get length of parsed device
338 000450 062701 ADD #6+3,R1 ; Add fixed overhead and round up
339 000454 042701 BIC #3,R1 ; Round it to the next 4 byte boundary
340 000460 060167 ADD R1,$BYTMN ; Add to minimum pool byte allocation
341
342 ; Point current field in FIL$DF template to file block
343
344 000464 016705 40$: MOV .TEMP,R5 ; Point to current template
345 000470 066705 ADD FILE,R5 ; Point to current file id
346 000474 010015 MOV R0,(R5) ; Point pointer to FD

```

SETDV1 - SETUP 1ST CHARACTER OF DEVICE NAME

```

157 .SBTTL SETDV1 - SETUP 1ST CHARACTER OF DEVICE NAME
158
159 :+
160 :*** - SETDV1 - SETUP 1ST CHARACTER OF DEVICE NAME
161 :
162 :INPUT:
163 :.PCHAR = THE 1ST CHARACTER OF THE DEVICE NAME
164 :TEMP = TEMPLATE ADDRESS
165 :
166 :OUTPUT:
167 :C.DEV = 1ST CHARACTER OF THE DEVICE NAME
168 :
169 :-
170 SETDV1:
171 000056 016700 177716 MOV TEMP,R0 ; GET START OF TEMPLATE
172 000062 116760 000000G 000004 MOVB .PCHAR,C.DEV(R0) ; STORE 1ST CHARACTER IN TEMPLATE
173 000070 000207 RETURN

```

SETDV2 - SETUP 2ND CHARACTER OF DEVICE NAME

```

135 000004 TRANS $NUMBR,EXT,SETNLT
136 000004 TRANS $LAMDA
137
138 000004 STATES$ EXT ; PROCESS EXTENSION
139 000004 TRANS $EOS,$EXIT
140 000004 TRANS <','>
141
142 000004 STATES$
143 000004 TRANS $NUMBR,CTM,SETEXT
144 000004 TRANS $LAMDA
145
146 000004 STATES$ CTM ; COUNTER TIMER
147 000004 TRANS $EOS,$EXIT
148 000004 TRANS <','>
149
150 000004 STATES$
151 000004 TRANS $NUMBR,,SETCTM
152
153 000004 STATES$
154 000004 TRANS $EOS,$EXIT
155 000004 TRANS $LAMDA,$EXIT
156
157 ;
158 ; SUB-EXPRESSION TO PARSE 'ZF.'"
159 ;
160 000004 STATES$ ZF
161 000004 TRANS "ZF"
162
163 000004 STATES$
164 000004 TRANS '.,$EXIT
165
166 ;
167 ; EXIT STATE
168 ;
168 000004 STATES$ EXIT
169 000004 TRANS $EOS,$EXIT
170
171 000004 STATES$

```

MACRO CROSS REFERENCE CREF 04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-82      | #5-95  | #5-102 | #5-103 | #5-104 | #5-105 | #5-106 | #5-107 | #5-108 | #5-124 |
|            | #5-128     | #5-132 | #5-136 | #5-140 | #5-144 | #5-148 | #5-155 |        |        |        |
| ISTAT\$    | #4-66      | 5-82   |        |        |        |        |        |        |        |        |
| LLCTP\$    | #4-66      | 4-68   |        |        |        |        |        |        |        |        |
| MSGDF\$    | #4-66      | 4-69   |        |        |        |        |        |        |        |        |
| MTRANS     | #5-82      |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-57      |        |        |        |        |        |        |        |        |        |
| STAT\$     | #4-66      | 5-84   | #5-87  | #5-90  | #5-93  | #5-97  | #5-100 | #5-110 | #5-113 | #5-116 |
|            | #5-119     | #5-122 | #5-126 | #5-130 | #5-134 | #5-138 | #5-142 | #5-146 | #5-150 | #5-153 |
|            | #5-160     | #5-163 | #5-168 | #5-171 |        |        |        |        |        |        |
| TRANS      | #4-66      | #5-85  | #5-88  | #5-91  | #5-94  | #5-95  | #5-98  | #5-101 | #5-102 | #5-103 |
|            | #5-104     | #5-105 | #5-106 | #5-107 | #5-108 | #5-111 | #5-114 | #5-117 | #5-120 | #5-123 |
|            | #5-124     | #5-127 | #5-128 | #5-131 | #5-132 | #5-135 | #5-136 | #5-139 | #5-140 | #5-142 |
|            | #5-144     | #5-147 | #5-148 | #5-151 | #5-154 | #5-155 | #5-161 | #5-164 | #5-169 |        |

IMPNOO - CFE PARSE NODE DEFINIT MACRO V05.03b Saturday 29-Jun-85 05:37 <sup>D 11</sup>  
Table of contents

|    |     |                               |
|----|-----|-------------------------------|
| 6- | 150 | \$TMNOD - SETUP NODE TEMPLATE |
| 7- | 181 | SETNAM - SET UP NODE NAME     |
| 8- | 215 | SETNID - SET UP NODE ID       |

TMPNOD      CREATED BY    MACRO    ON 29-JUN-85 AT 05:37      PAGE 2      D 12  
 MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |  |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--|
| DBGTP\$ | #5-99  | #5-146 |        |        |        |        |        |        |        |        |  |
| ISTAT\$ | #4-68  | 5-99   |        |        |        |        |        |        |        |        |  |
| MTRANS  | #5-99  |        |        |        |        |        |        |        |        |        |  |
| PRINT   | #4-59  |        |        |        |        |        |        |        |        |        |  |
| STAT\$  | #4-68  | 5-102  | #5-105 | #5-108 | #5-111 | #5-114 | #5-117 | #5-120 | #5-123 | #5-126 |  |
|         | #5-129 | #5-132 | #5-135 | #5-138 | #5-141 | #5-144 | #5-148 |        |        |        |  |
| TRANS   | #4-68  | #5-103 | #5-106 | #5-109 | #5-112 | #5-115 | #5-118 | #5-121 | #5-124 | #5-127 |  |
|         | #5-130 | #5-133 | #5-136 | #5-139 | #5-142 | #5-145 | #5-146 |        |        |        |  |



TMPOBJ ~ CFE PARSE OBJECT DEFIN MACRO V05.03b Saturday 29-Jun-85 05:37 Page 12-1

```

Symbol table
CS.OBJ= ***** GX OBJKT B 000000RG 003 TEMP 000000R $EXIT = 000000 $T1ALC= ***** GX
C.CPY 000012 OBJST B 000000RG 002 $ALPHA= 000022 $FAIL = 177777 $$$FLG= 177777
C.FLG 000005 OBJJ 000024R 002 $ANY = 000020 $LAMBDA= 000000 $$$KEY= 177777
C.LEN = 000013 D.LENX= ***** GX $BLANK= 000006 $NUMBR= 000002 $$$STA= 000000
C.NAM 000006 RA = 000076 $BYTMN= ***** GX $QINSN= ***** GX .PNUMB= ***** GX
C.STS = ***** GX SETCPY 000366R $CAT5 = ***** GX $RAD50= 000016 .PNUMH= ***** GX
C.TYP 000004 SETFLG 000322R $DIGIT= 000024 $STRNG= 000004 .PSTCN= *** ** GX
EXIT 000050R 002 SETNAM 000206R $DNUMB= 000014 $SUBXP= 000010 .PSTPT= ***** GX
FINOBJ 000116R SETVFX 000264R $EOS = 000012 $TMLST= ***** GX .TPARS= ***** GX
LA = 000074 SETYPE 000152R $ERROR= ***** GX $TMDBJ 000002RG

```

```

. ABS. 000013 000 (RW,I,GBL,ABS,OVR)
 000424 001 (RW,I,LCL,REL,CON)
$STATE 000054 002 (RW,D,LCL,REL,CON)
$KTAB 000000 003 (RW,D,LCL,REL,CON)
$KSTR 000000 004 (RW,D,LCL,REL,CON)
Errors detected: 0

```

\*\*\* Assembler statistics

```

Work file reads: 0
Work file writes: 0
Size of work file: 10406 Words (41 Pages)
Size of core pool: 14440 Words (55 Pages)
Operating system: RSX-11M/PLUS

```

```

Elapsed time: 00:00:23.22
SY: TMPOBJ.V2,[132,134]TMPOBJ/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPOBJ

```

277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307

.SBTTL SETALL - SET THE BYTE-AREA EXTENSION

\*\*\* - SETALL - SET THE POOL BYTE-AREA EXTENSION

INPUT:

.PNUMB = BYTE-AREA EXTENSION  
 .PNUMH = BYTE-AREA EXTENSION (HIGH ORDER)  
 TEMP = TEMPLATE ADDRESS

OUTPUT:

THE BYTE-AREA EXTENSION IS STORED IN C.ALL OF THE TEMPLATE.

\*\*NOTE\*\* IF THE MAXIMUM VALUE GOES OVER 1024. BLOCKS \$BYTMN AND \$BYTXT  
 WILL HAVE TO BECOME DOUBLE PRECISION.

SETALL:

MOV TEMP,RO ; GET START OF TEMPLATE  
 MOV .PNUMB,C.ALL(RO) ; STORE THE BYTE-AREA EXTENSION IN THE TEMPLATE  
 TST .PNUMH ; IS IT A WORD VALUE ?  
 BNE 10\$ ; IF NE, NO - REJECT  
 CMP .PNUMB,#1023. ; IS THE BYTE-AREA EXTENSION IN RANGE ?  
 BLOS 20\$ ; IF LOS, YES - OKAY  
 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION  
 BR 30\$ ; JOIN COMMON CODE  
 MOV C.ALL(RO),RO ; GET ALLOCATION VALUE IN 32.WD BLOCKS  
 .REPT 6  
 ASI RO ; \*64. TO GET VALUE IN BYTES  
 .ENDR  
 MOV RO,\$BYTXT ; SAVE IT FOR AUTO EXTEND LOGIC  
 RETURN

10\$:  
20\$:  
30\$:

000272 016700 177504  
 000272 016760 000000G 000016  
 000304 005767 000000G  
 000310 001004  
 000312 026727 000000G 001777  
 000320 101403  
 000322 062716 000002  
 000326 000412  
 000330 016000 000016  
 000006  
 000350 010067 000000G  
 000354 000207

```

214 .SBTTL $TMPSN - BUILD PSN TEMPLATE
215
216 *** - $TMPSN - BUILD PSN TEMPLATE
217
218 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
219
220 INPUT:
221 R3-R5 - TPARS REGISTERS
222 $PSIPT - POINTER TO PREVIOUS PSN$DF IF ANY
223
224 OUTPUT:
225 C-BIT = SUCCESS/FAILURE
226 $PSIPT POINTS TO PSN TEMPLATE
227 THE PSN TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
228
229
230
231 $TMPSN::
232 TST $PSIPT ; TEST FOR PREVIOUS PSN$DF
233 BNE 101$; IF NE, EXIT
234 MOV R5, -(SP) ; SAVE R5
235 MOV #C.LEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
236 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
237 MOV RO, .TEMP ; SAVE TEMPLATE ADDRESS
238 MOV RO, $PSIPT ; SAVE FOR LATER CHECKS
239 TST (R0)+ ; POINT TO C.STS
240 MOV RO, .FLAGS ; SAVE POINTER FOR OPTION SETTING
241 MOV #CS.PSN, (R0) ; INDICATE THAT THIS IS A PSN$DF TEMPLATE
242 CLR .ERROR ; NO DETAIL STRING YET
243 CLR R1 ; IGNORE BLANKS
244 MOV #PSNKT.B, R2 ; KEYWORD TABLE
245 MOV #PSNST.B, R5 ; STATE TABLE
246 CALL .TPARS ; PARSE THE REST OF THE LINE
247 BCC 20$; NORMAL RETURN IF NO ERROR
248 MOV .ERROR, R0 ; GET DETAIL STRING
249 BEQ 15$; IF EQ, NO ERROR
250 ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
251 MOV #-2, $ERROR ; INDICATE ERROR AND MESSAGE ALREADY PRINTED
252 INC $ERROR ; INDICATE SYNTAX ERROR
253 MOV (SP)+, R5 ; RESTORE R5
254 RETURN
255
256 101$: ERROR$ #ERRPSN ; DUPLICATE PSN$DF
257 DEC $ERROR ; INDICATE SYNTAX ERROR
258 RETURN
259
260 .SBTTL SETNAM - SET PSN NAME IN TEMPLATE
261 .SBTTL SETPOR - SET NUMBER OF PORTS IN TEMPLATE
262 .SBTTL SETFLG - SET FLAGS IN TEMPLATE
263 .SBTTL SETTHI - SET TRANSPORT HIGH ADDRESS IN TEMPLATE
264 .SBTTL SETTLO - SET TRANSPORT LOW ADDRESS IN TEMPLATE
265
266
267 ** SETNAM - SET PSN NAME IN TEMPLATE
268 ** SETPOR - SET NUMBER OF PORTS IN TEMPLATE
269 ** SETFLG - SET FLAGS IN TEMPLATE
270 ** SETTHI - SET TRANSPORT HIGH ADDRESS IN TEMPLATE

```

```

206 000354 TRANS $EOS,$EXIT
207 000354 TRANS <'>
208
209 000354 STATES
210 000354 TRANS $LAMDA,, $CLERR,ERROWN,.ERROR ; OPTIONAL OWNER PROCESS
211 000354 STATES
212 000354 TRANS $RAD50,PVC4,SETOWN
213 000354 TRANS $LAMDA
214
215 000354 STATES PVC4
216 000354 TRANS $EOS,$EXIT
217 000354 TRANS <'>
218
219 000354 STATES
220 000354 TRANS $LAMDA,, $CLERR,ERRFLG,.ERROR ; OPTIONAL FLAGS BYTE
221 000354 STATES
222 000354 TRANS $NUMBR,,SETFLG
223 000354 STATES
224 000354 TRANS $EOS,$EXIT
225
226 ; SUBEXPRESSION TO SPECIAL PVC NAMES FOR DLM
227 ;
228
229 000354 STATES DLMPVC ;DEVICE NAME
230 000354 TRANS "DLM",,SETDEV
231 000354 STATES
232 000354 TRANS <'>
233 000354 STATES ;CONTROLLER NUMBER
234 000354 TRANS $DNUMB,,SETCON
235 000354 STATES
236 000354 TRANS <'>
237 000354 STATES ;TRIBUTARY NUMBER
238 000354 TRANS $DNUMB,$EXIT,SETTRB
239
240 ;
241 ; FINAL STATE
242 ;
243 000354 STATES
244

```

TMPDLC CREATED BY MACRO ON 29-JUN-85 AT 05:31 PAGE 1 E 1  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL    | VALUE       | REFERENCES                                   |
|-----------|-------------|----------------------------------------------|
| CS.DLC    | = ***** GX  | 6-170                                        |
| C.FLG     | = 000006    | #4-84 *8-222                                 |
| C.LEN     | = 000014    | #4-88 6-167                                  |
| C.NAM     | = 000004    | #4-83 *7-201                                 |
| C.NCT     | = 000013    | #4-87 *11-285                                |
| C.NLT     | = 000012    | #4-86 *10-265                                |
| C.PRI     | = 000010    | #4-85 *9-243                                 |
| C.STS     | = ***** GX  | *6-170                                       |
| DLCKTB    | = 000000 RG | #5-96 6-173                                  |
| DLCSTB    | = 000000 RG | #5-96 6-174                                  |
| FLAG      | = 000000 R  | #4-74 8-222                                  |
| SETFLG    | = 000124 R  | #8-220                                       |
| SETNAM    | = 000064 R  | #7-196                                       |
| SETNCT    | = 000204 R  | #11-283                                      |
| SETNLT    | = 000154 R  | #10-263                                      |
| SETPRI    | = 000140 R  | #9-241                                       |
| TEMP      | = 000002 R  | #4-75 *6-169 7-200 8-221 9-242 10-264 11-284 |
| \$ALPHA   | = 000022    | #5-96                                        |
| \$ANY     | = 000020    | #5-96                                        |
| \$BLANK   | = 000006    | #5-96                                        |
| \$CAT5    | = ***** GX  | 7-199                                        |
| \$DIGIT   | = 000024    | #5-96                                        |
| \$DNUMB   | = 000014    | #5-96                                        |
| \$EOS     | = 000012    | #5-96                                        |
| \$ERROR   | = ***** GX  | *6-177                                       |
| \$EXIT    | = 000000    | #5-96                                        |
| \$FAIL    | = 177777    | #5-96                                        |
| \$GPRM    | = *****     | 5-96                                         |
| \$LAMDA   | = 000000    | #5-96                                        |
| \$MXNCT   | = ***** GX  | 11-286                                       |
| \$MXNLT   | = ***** GX  | 10-266                                       |
| \$NUMBR   | = 000002    | #5-96                                        |
| \$RAD50   | = 000016    | #5-96                                        |
| \$RONLY   | = *****     | 5-96 5-96                                    |
| \$STRNG   | = 000004    | #5-96                                        |
| \$SUBXP   | = 000010    | #5-96                                        |
| \$TALOC   | = ***** GX  | 6-168                                        |
| \$TMDLC   | = 000004 RG | #6-165                                       |
| \$\$\$FLG | = 177777    | #5-96                                        |
| \$\$\$KEY | = 177777    | #5-96                                        |
| .PNUMB    | = ***** GX  | 9-243 10-265 10-266 11-285 11-286            |
| .PSTCN    | = ***** GX  | 7-202                                        |
| .PSTPT    | = ***** GX  | 7-197                                        |
| .TPARS    | = ***** GX  | 6-175                                        |

MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| CALLR      | #4-124     |        |        |        |        |        |        |        |        |        |
| DBGTP\$    | #6-145     | #6-152 | #6-155 | #6-159 | #6-162 | #6-166 | #6-169 | #6-173 | #6-185 | #6-188 |
|            | #6-189     |        |        |        |        |        |        |        |        |        |
| EPRINT     | #4-78      |        |        |        |        |        |        |        |        |        |
| ERROR\$    | #4-67      | 7-227  |        |        |        |        |        |        |        |        |
| ISTAT\$    | #4-124     | 6-145  |        |        |        |        |        |        |        |        |
| MTRANS     | #6-145     |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-62      |        |        |        |        |        |        |        |        |        |
| STATES     | #4-124     | 6-150  | #6-153 | #6-157 | #6-160 | #6-164 | #6-167 | #6-171 | #6-174 | #6-177 |
|            | #6-183     | #6-186 | #6-193 |        |        |        |        |        |        |        |
| TRANS      | #4-124     | #6-151 | #6-152 | #6-154 | #6-155 | #6-158 | #6-159 | #6-161 | #6-162 | #6-165 |
|            | #6-166     | #6-168 | #6-169 | #6-172 | #6-173 | #6-175 | #6-178 | #6-184 | #6-185 | #6-187 |
|            | #6-188     | #6-189 |        |        |        |        |        |        |        |        |
| \$GTNMB    | #4-100     |        |        |        |        |        |        |        |        |        |
| \$GTNUM    | #4-90      | 8-257  | 8-259  |        |        |        |        |        |        |        |
| \$GTR50    | #4-112     | 8-255  | 8-261  |        |        |        |        |        |        |        |

\*\*FILE\*\*ID\*\*TMPDST

|           |     |     |         |          |          |           |
|-----------|-----|-----|---------|----------|----------|-----------|
| TTTTTTTTT | MM  | MM  | PPPPPPP | DDDDDDDD | SSSSSSSS | TTTTTTTTT |
| TTTTTTTTT | MM  | MM  | PPPPPPP | DDDDDDDD | SSSSSSSS | TTTTTTTTT |
| TT        | MMM | MMM | PP      | DD       | SS       | TT        |
| TT        | MMM | MMM | PP      | DD       | SS       | TT        |
| TT        | MM  | MM  | PP      | DD       | SS       | TT        |
| TT        | MM  | MM  | PP      | DD       | SS       | TT        |
| TT        | MM  | MM  | PPPPPPP | DD       | SSSSSS   | TT        |
| TT        | MM  | MM  | PPPPPPP | DD       | SSSSSS   | TT        |
| TT        | MM  | MM  | PP      | DD       | SS       | TT        |
| TT        | MM  | MM  | PP      | DD       | SS       | TT        |
| TT        | MM  | MM  | PP      | DD       | SS       | TT        |
| TT        | MM  | MM  | PP      | DD       | SS       | TT        |
| TT        | MM  | MM  | PP      | DD       | SS       | TT        |
| TT        | MM  | MM  | PP      | DDDDDDDD | SSSSSSSS | TT        |
| TT        | MM  | MM  | PP      | DDDDDDDD | SSSSSSSS | TT        |

....  
 ....  
 ....  
 ....

|            |          |           |
|------------|----------|-----------|
| LL         | SSSSSSSS | TTTTTTTTT |
| LL         | SSSSSSSS | TTTTTTTTT |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SSSSSS   | TT        |
| LL         | SSSSSS   | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LLLLLLLLLL | SSSSSSSS | TT        |
| LLLLLLLLLL | SSSSSSSS | TT        |

TMPDTE - PARSE LOCAL DTE DESCR MACRO V05.03b Saturday 29-Jun-85 05:33 <sup>F 4</sup>  
Table of contents

|    |     |                                           |
|----|-----|-------------------------------------------|
| 4- | 57  | MACRO DEFINITIONS                         |
| 5- | 127 | LOCAL DATA                                |
| 6- | 159 | TPARS STATE TABLES                        |
| 7- | 243 | \$TMDTE - BUILD DTE TEMPLATE              |
| 8- | 283 | SETADR - SET DTE ADDRESS IN TEMPLATE      |
| 8- | 284 | SETHSH - SET HASH TABLE SIZE IN DTE       |
| 8- | 285 | SETCT - SET COUNTER TIMER VALUE           |
| 8- | 286 | SETNET - SET NETWORK NAME IN DTE TEMPLATE |
| 8- | 319 | SETSTA - SET DTE STATE IN TEMPLATE        |
| 8- | 333 | SETDEV - SET DEVICE NAME IN LINE ID       |
| 8- | 334 | SETCON - SET CONTROLLER NUMBER IN LINE ID |
| 8- | 335 | SETUNI - SET UNIT NUMBER IN LINE ID       |
| 8- | 336 | SETTRB - SET TRIB NUMBER IN LINE ID       |



TMPDTE      CREATED BY    MACRO    ON 29-JUN-85 AT 05:33      PAGE 3      E 5  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES                                                                                       |
|------------|--------------------------------------------------------------------------------------------------|
| CALLR      | #4-124                                                                                           |
| DBGTP\$    | #6-164    #6-197    #6-200    #6-216    #6-228    #6-229    #6-234                               |
| EPRINT     | #4-78                                                                                            |
| ERROR\$    | #4-67    7-276                                                                                   |
| ISTAT\$    | #4-124    6-164                                                                                  |
| MTRANS     | #6-164                                                                                           |
| PRINT      | #4-62                                                                                            |
| STAT\$     | #4-124    6-169    #6-171    #6-173    #6-177    #6-179    #6-183    #6-185    #6-189    #6-191  |
|            | #6-195    #6-198    #6-202    #6-204    #6-206    #6-212    #6-214    #6-220    #6-222    #6-224 |
|            | #6-226    #6-230    #6-232    #6-235    #6-240                                                   |
| TRANS      | #4-124    #6-170    #6-172    #6-174    #6-178    #6-180    #6-184    #6-186    #6-190    #6-192 |
|            | #6-196    #6-197    #6-199    #6-200    #6-203    #6-205    #6-207    #6-213    #6-215    #6-216 |
|            | #6-221    #6-223    #6-225    #6-227    #6-228    #6-229    #6-231    #6-233    #6-234    #6-236 |
| \$GTNMB    | #4-100                                                                                           |
| \$GTNUM    | #4-90    8-310    8-315    8-350    8-352    8-354                                               |
| \$GTR50    | #4-112    8-298    8-317    8-348                                                                |

TMPEVT      CREATED BY    MACRO    ON 29-JUN-85 AT 05:34      PAGE 1      E    6

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE      | REFERENCES                                |
|---------|------------|-------------------------------------------|
| CS.EVT  | = ***** GX | 7-163 8-198                               |
| C.CLS   | 000004     | #5-92 8-200                               |
| C.EVT   | 000006     | #5-93 *7-164 8-200 *9-225                 |
| C.LEN   | = 000020   | #5-96 *7-165 *7-166 *7-167 9-229 10-253   |
| C.SNK   | 000016     | #5-94 8-202 8-202 *11-294                 |
| C.STS   | = ***** GX | *7-163 8-198                              |
| EVTCTB  | 000000 RG  | #6-107 7-169                              |
| EVTSTB  | 000000 RG  | #6-107 7-170                              |
| FF.ADD  | = 000020   | #4-72 4-72                                |
| FF.CIR  | = 000040   | #4-72 4-72                                |
| FF.CON  | = 000001   | #4-72 11-283                              |
| FF.FIL  | = 000002   | #4-72 11-291                              |
| FF.HST  | = 040000   | #4-72                                     |
| FF.LIN  | = 000010   | #4-72 4-72                                |
| FF.MOD  | = 000100   | #4-72 4-72                                |
| FF.MON  | = 000004   | #4-72 11-287                              |
| FF.MSK  | = 000077   | #4-72                                     |
| FF.PRT  | = 000200   | #4-72 4-72                                |
| FF.QUL  | = 000370   | #4-72                                     |
| FF.REM  | = 100000   | #4-72                                     |
| FINEVT  | 000122 R   | 7-175 #8-193                              |
| F.ADD   | 000016     | #4-72                                     |
| F.CEV   | 000030     | #4-72                                     |
| F.CIR   | 000024     | #4-72                                     |
| F.CLS   | 000002     | #4-72                                     |
| F.EVT   | 000004     | #4-72                                     |
| F.FLG   | 000014     | #4-72                                     |
| F.LEN   | 000016     | #4-72                                     |
| F.LENX  | = ***** GX | 7-177                                     |
| F.LIN   | 000020     | #4-72                                     |
| F.LNK   | 000000     | #4-72                                     |
| F.MOD   | 000022     | #4-72                                     |
| F.REM   | 000026     | #4-72                                     |
| F.SEV   | 000004     | #4-72                                     |
| LBRA    | = 000074   | #5-80                                     |
| MAXCLS  | = 077700   | #5-82 9-223                               |
| RBRA    | = 000076   | #5-81                                     |
| SETCLS  | 000166 R   | #9-221                                    |
| SETCON  | 000310 R R | #11-283                                   |
| SETFIL  | 000324 R R | #11-291                                   |
| SETMON  | 000316 R R | #11-287                                   |
| SETMSK  | 000240 R R | #10-249                                   |
| STRADD  | 000002 R   | #5-84 *9-228 *9-229 10-254 10-256 *10-257 |
| TEMP    | 000000 R   | #5-83 *7-162 7-174 9-222 10-252 11-293    |
| \$ALPHA | = 000022   | #6-107                                    |
| \$ANY   | = 000020   | #6-107                                    |
| \$BLANK | = 000006   | #6-107                                    |
| \$BYTMN | = ***** GX | *7-177                                    |
| \$DIGIT | = 000024   | #6-107                                    |
| \$DNUMB | = 000014   | #6-107                                    |
| \$EOS   | = 000012   | #6-107                                    |
| \$ERROR | = ***** GX | *7-173                                    |

TMPFIL - CFE PARSE REMOTE CHARA MACRO V05.03b Saturday 29-Jun-85 05:34 Page 7-2  
Action routines

347 000476 000207  
348  
349

RETURN

SETDV2 - SETUP 2ND CHARACTER OF DEVICE NAME

```

175 .SBTTL SETDV2 - SETUP 2ND CHARACTER OF DEVICE NAME
176
177 :+
178 *** - SETDV2 - SETUP 2ND CHARACTER OF DEVICE NAME
179 :
180 INPUT:
181 .PCHAR = THE 2ND CHARACTER OF THE DEVICE NAME
182 TEMP = TEMPLATE ADDRESS
183 :
184 OUTPUT:
185 C.DEV+1 = 2ND CHARACTER OF THE DEVICE NAME
186 :-
187
188 000072 SETDV2:
189 000072 016700 177702 MOV TEMP,R0 ; GET START OF TEMPLATE
190 000076 116760 000000G 000005 MOVB .PCHAR,C.DEV+1(R0) ; STORE 2ND CHARACTER IN TEMPLATE
191 000104 000207 RETURN

```

SETWD1 - SETUP INITIAL FEATURES WORD

```

173 .SBTTL $TMLLC - SETUP LLC TEMPLATE
174 ;+
175 *** - $TMLLC - SETUP LLC TEMPLATE
176 ;
177 INPUT:
178 R3-R5 - TPARS REGISTERS
179 ;
180 OUTPUT:
181 C-BIT = SUCCESS/FAILURE
182 IF SUCCESS, THE LLC TEMPLATE IS STORED IN THE END OF TASK BUFFER.
183 ;
184 :-
185
186 000004 $TMLLC::
187 000004 010546 MOV R5,-(SP) ; SAVE R5
188 000006 005067 177766 CLR FLAG ; CLEAR THE FLAGS BYTE
189 000012 012701 000^20 MOV #CL.LEN,R1 ; GET LENGTH OF ALLOCATION
190 000016 004767 000^30G CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
191 000022 010067 177754 MOV R0,TEMP ; STORE TEMPLATE ADDRESS
192 000026 012760 000000G 000000G MOV #CS.LLC.C.STS(R0) ; INDICATE THAT THIS IS AN LLC$DF TEMPLATE
193 000034 062700 000014 ADD #C.NLT,R0 ; POINT TO OPTIONAL PARAMETERS
194 000040 005020 CLR (R0)+ ; CLEAR OPTIONAL PARAMETERS
195 000042 005020 CLR (R0)+ ;
196 000044 005001 CLR R1 ; IGNORE BLANKS
197 000046 012702 000000' MOV #LLCKTB,R2 ; KEYWORD TABLE
198 000052 012705 000000' MOV #LLCSTB,R5 ; STATE TABLE
199 000056 004767 000000G CALL ,TPARS ; PARSE THE REST OF THE LINE
200 000062 103002 BCC 20$; NORMAL RETURN IF NO ERROR
201 000064 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
202 000070 012605 20$: MOV (SP)+,R5 ; RESTORE R5
203 000072 000207 RETURN

```

[illegible]

```

LL SSSSSSSS TTTTTTTTTTTT
LL ScSSSSS TTTTTTTTTTTT
LL Sc TT
LL Sc TT
LL Sc TT
LL Sc TT
LL Sc TT
LL SSSSSS TT
LL SSSSSS TT
LL SSS SS TT
LL SSS SS TT
LL SSS SS TT
LL SSS SS TT
LLLLLLLLLLLL SSSSSSSS TT
LLLLLLLLLLLL SSSSSSSS TT

```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53

..TITLE TMPNOD - CFE PARSE NODI DEFINITION IN CETAB  
..IDENT /V05.00/

..COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
..DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

..THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
..ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
..INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
..COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
..OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
..TRANSFERRED.

..THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
..AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
..CORPORATION.

..DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
..SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

..MODULE DESCRIPTION:

..STATE TABLE TO PARSE THE NODE DEFINITIONS

..DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

..IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M v4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

|          |      |      |         |        |          |          |       |
|----------|------|------|---------|--------|----------|----------|-------|
| TTTTTTTT | MM   | MM   | PPPPPPP | 000000 | 88888888 |          | JJ    |
| TTTTTTTT | MM   | MM   | PPPPPPP | 000000 | 88888888 |          | JJ    |
| TT       | MMMM | MMMM | PP      | 00     | 00       | 88       | JJ    |
| TT       | MMMM | MMMM | PP      | 00     | 00       | 88       | JJ    |
| TT       | MM   | MM   | PP      | 00     | 00       | 88       | JJ    |
| TT       | MM   | MM   | PP      | 00     | 00       | 88       | JJ    |
| TT       | MM   | MM   | PPPPPPP | 00     | 00       | 88888888 | JJ    |
| TT       | MM   | MM   | PPPPPPP | 00     | 00       | 88888888 | JJ    |
| TT       | MM   | MM   | PP      | 00     | 00       | 88       | JJ    |
| TT       | MM   | MM   | PP      | 00     | 00       | 88       | JJ    |
| TT       | MM   | MM   | PP      | 00     | 00       | 88       | JJ    |
| TT       | MM   | MM   | PP      | 00     | 00       | 88       | JJ    |
| TT       | MM   | MM   | PP      | 00     | 00       | 88       | JJ    |
| TT       | MM   | MM   | PP      | 00     | 00       | 88       | JJ    |
| TT       | MM   | MM   | PP      | 000000 | 88888888 | JJJJJJ   | ..... |
| TT       | MM   | MM   | PP      | 000000 | 88888888 | JJJJJJ   | ..... |

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
 SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSSSS TT
LLLLLLLLLLL SSSSSSSS TT
LLLLLLLLLLL SSSSSSSS TT

```



IMPOBJ CREATED BY MACRO ON 29-JUN-85 AT 05:37 PAGE 1 E 13  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL  | VALUE       | REFERENCES                                                   |
|---------|-------------|--------------------------------------------------------------|
| CS.OBJ  | = ***** GX  | 6-164 7-198                                                  |
| C.CPY   | = 000012    | #4-90 *6-167 *12-331                                         |
| C.FLG   | = 000005    | #4-88 *10-279 *11-304                                        |
| C.LEN   | = 000013    | #4-92 6-161                                                  |
| C.NAM   | = 000006    | #4-89 *6-165 *6-166 *9-250 *9-254                            |
| C.STS   | = ***** GX  | *6-164 7-198                                                 |
| C.TYP   | = 000004    | #4-87 7-200 7-200 *8-221                                     |
| FINOBJ  | = 000116 R  | 6-175 #7-193                                                 |
| LA      | = 000074    | #4-77                                                        |
| OBJKTG  | = 000000 RG | #5-101 6-169                                                 |
| OBJSTG  | = 000000 RG | #5-101 6-170                                                 |
| O.LENX  | = ***** GX  | 6-177                                                        |
| RA      | = 000076    | #4-78                                                        |
| SETCPY  | = 000366 R  | #12-329                                                      |
| SETFLG  | = 000322 R  | #11-302                                                      |
| SETNAM  | = 000206 R  | #9-245                                                       |
| SETVfy  | = 000264 R  | #10-277                                                      |
| SETYPE  | = 000152 R  | #8-219                                                       |
| TEMP    | = 000000 R  | #4-79 *6-163 6-174 8-220 9-249 9-253 10-278 11-303 12-330    |
| \$ALPHA | = 000022    | #5-101                                                       |
| \$ANY   | = 000020    | #5-101                                                       |
| \$BLANK | = 000006    | #5-101                                                       |
| \$BYTMN | = ***** GX  | *6-177                                                       |
| \$CAT5  | = ***** GX  | 9-252                                                        |
| \$DIGIT | = 000024    | #5-101                                                       |
| \$DNUMB | = 000014    | #5-101                                                       |
| \$EOS   | = 000012    | #5-101                                                       |
| \$ERROR | = ***** GX  | *6-173                                                       |
| \$EXIT  | = 000000    | #5-101                                                       |
| \$FAIL  | = 177777    | #5-101                                                       |
| \$GPRM  | = *****     | 5-101                                                        |
| \$LAMDA | = 000000    | #5-101                                                       |
| \$NUMBR | = 000002    | #5-101                                                       |
| \$QINSN | = ***** GX  | 6-176                                                        |
| \$RAD50 | = 000016    | #5-101                                                       |
| \$RONLY | = *****     | 5-101 5-101                                                  |
| \$STRNG | = 000004    | #5-101                                                       |
| \$SUBXP | = 000010    | #5-101                                                       |
| \$TMLST | = ***** GX  | 7-194                                                        |
| \$TMOBJ | = 000002 RG | #6-159                                                       |
| \$TIALC | = ***** GX  | 6-162                                                        |
| \$SFLG  | = 177777    | #5-101                                                       |
| \$SKEY  | = 177777    | #5-101                                                       |
| .PNUMB  | = ***** GX  | 8-221 8-222 10-279 10-282 11-304 11-307 11-309 12-331 12-334 |
| .PNUMH  | = ***** GX  | 10-280 11-305 12-335                                         |
| .STCN   | = ***** GX  | 9-255                                                        |
| .PSYPT  | = ***** GX  | 9-246                                                        |
| .TPARS  | = ***** GX  | 6-171                                                        |

TMPPAR - CFE PARSE PARTITION DE MACRO V05.03b Saturday 29-Jun-85 05:38 Page 11  
 SETFLG - SETUP THE FLAGS WORD

```

309 .SBTTL SETFLG - SETUP THE FLAGS WORD
310
311 +
312 *** - SETFLG - SETUP THE FLAGS WORD
313 :
314 INPUT:
315 FLAG = THE FLAGS WORD
316 TEMP = START ADDRESS OF THE TEMPLATE
317 :
318 OUTPUT:
319 :
320 -
321
322 000356 SETFLG:
323 000356 016700 177420 MOV TEMP,R0 ; GET START OF TEMPLATE
324 000362 056760 177412 000000G BIS FLAG,C.STS(R0) ; STORE THE FLAGS WORD
325 000370 000207 RETURN
326
327 .END

```

```

271 ; ** SETTLO - SET TRANSPORT LOW ADDRESS IN TEMPLATE
272 ;
273 ; INPUT: .TPARS CONVENTIONS
274 ;
275 ; OUTPUT: VALUE SET IN TEMPLATE
276 ;
277 ; -
278
279 000362 SETNAM: $GTR50 #C.NAM,#2
280
281 000376 016700 000000G SETPOR: MOV .PNUMB,R0 ; SIZE IF PORT TABLE
282 000402 005200 INC R0 ; OVERHEAD OF ONE
283 000404 006300 ASL R0 ; CONVERT TO BYTE VALUE
284 000406 062700 000003 ADD #3,R0 ; ON 4 BYTE BOUNDARY
285 000412 042700 000003 BIC #3,R0
286 000416 060067 000000G ADD R0,$BYTMN ; UPDATE MINIMUM ALLOCATION
287 000422 $GTNUM #C.POR,#1,#256.
288
289 000442 SETFLG: $GTNUM #C.FLG,,#CS.FLG
290
291 000470 SETTHI: $GTNUM #C.THI,,#9999.,#CS.THI
292
293 000516 SETTLO: $GTNUM #C.TLO,,#9999.,#CS.TLO
294
295 000001 .END

```

```

246 .SETTL $TMPVC - BUILD PVC TEMPLATE
247
248 ;+ *** - $TMPVC - BUILD PVC TEMPLATE
249
250 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
251
252 INPUT:
253 R3-R5 - TPARS REGISTERS
254
255 OUTPUT:
256 C-BIT = SUCCESS/FAILURE
257 THE PVC TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
258
259 -
260
261 $TMPVC::
262 MOV R5, -(SP) ; SAVE R5
263 MOV #P$VLEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
264 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
265 MOV RO, TEMP ; SAVE TEMPLATE ADDRESS
266 TST (R0)+ ; POINT TO C.STS
267 MOV RO, .FLAGS ; SAVE FOR OPTION SETTING
268 MOV #CS.PVC, (R0) ; INDICATE THAT THIS IS A PVC$DF TEMPLATE
269 CLR .ERROR ; NO ERROR DETAIL YET
270 STA ; ASSUME STATE=OFF
271 CLR DLM ; ASSUME NORMAL PVC NAME
272 ADD #X$LENX, $BYTMN ; UPDATE MINIMUM BYTE ALLOCATION
273 CLR R1 ; IGNORE BLANKS
274 MOV #PVCKTB, R2 ; KEYWORD TABLE
275 MOV #PVCSTB, R5 ; STATE TABLE
276 CALL .TPARS ; PARSE THE REST OF THE LINE
277 BCS 10$; IF CS, ERROR IN PARSE
278 DLM 20$; WAS IT A SPECIAL DLM NAME?
279 BEQ 20$; IF EQ, NO - JUST RETURN
280 ADD #C$LENX, $BYTMN ; ADD NAME BLOCK LENGTH TO MINIMUM
281 MOV .TEMP, R1 ; POINT TO TEMPLATE
282 CMF P$VOWN(R1), #^RDLM ; ONLY VALID IF OWNER=DLM
283 BEQ 20$; BR IF VALID OWNER
284 MOV #ERRNAM, .ERROR ; ELSE ILLEGAL NAME
285 MOV .ERROR, R0 ; CHECK FOR ERROR DETAIL TEXT
286 BEQ 15$; IF EQ, NO DETAIL TEXT
287 ERROR$ RO ; PRINT ACTION ROUTINE ERROR
288 MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
289 INC $ERROR ; INDICATE SYNTAX ERROR
290 MOV (SP)+, R5 ; RESTORE R5
291 RETURN
292
293 .SBTTL SETNAM - SET PVC NAME
294 .SBTTL SETLCN - SET LOGICAL CHANNEL NUMBER
295 .SBTTL SETCT - SET COUNTER TIMER
296 .SBTTL SETPSZ - SET MAXIMUM BLOCK SIZE
297 .SBTTL SETWSZ - SET WINDOW SIZE
298 .SBTTL SETOWN - SET OWNER PROCESS
299 .SBTTL SETFLG - SET FLAGS BYTE
300 .SBTTL SETSTA - SET PVC STATE
301
302 ;+ ** SETNAM - SET PVC NAME

```

TMPDLC      CREATED BY    MACRO    ON 29-JUN-85 AT 05:31      Page 2      F 1  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-96      | #5-112 | #5-119 | #5-120 | #5-121 | #5-131 | #5-135 | #5-139 |        |        |
| ISTAT\$    | #4-68      | 5-96   |        |        |        |        |        |        |        |        |
| MTRANS     | #5-96      |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-59      |        |        |        |        |        |        |        |        |        |
| STATE\$    | #4-68      | 5-98   | #5-101 | #5-104 | #5-107 | #5-110 | #5-114 | #5-117 | #5-123 | #5-126 |
|            | #5-129     | #5-133 | #5-137 | #5-141 | #5-144 | #5-147 | #5-150 |        |        |        |
| TRANS      | #4-68      | #5-99  | #5-102 | #5-105 | #5-108 | #5-111 | #5-112 | #5-115 | #5-118 | #5-119 |
|            | #5-120     | #5-121 | #5-124 | #5-127 | #5-130 | #5-131 | #5-134 | #5-135 | #5-138 | #5-139 |
|            | #5-142     | #5-145 | #5-148 |        |        |        |        |        |        |        |

|            |      |      |          |          |          |          |
|------------|------|------|----------|----------|----------|----------|
| TTTTTTTTTT | MM   | MM   | PPPPPPPP | DDDDDDDD | SSSSSSSS | CCCCCCCC |
| TTTTTTTTTT | MM   | MM   | PPPPPPPP | DDDDDDDD | SSSSSSSS | CCCCCCCC |
| TT         | MMMM | MMMM | PP       | DD       | SS       | CC       |
| TT         | MMMM | MMMM | PP       | DD       | SS       | CC       |
| TT         | MM   | MM   | PP       | DD       | SS       | CC       |
| TT         | MM   | MM   | PP       | DD       | SS       | CC       |
| TT         | MM   | MM   | PPPPPPPP | DD       | SSSSSS   | CC       |
| TT         | MM   | MM   | PPPPPPPP | DD       | SSSSSS   | CC       |
| TT         | MM   | MM   | PP       | DD       | SS       | CC       |
| TT         | MM   | MM   | PP       | DD       | SS       | CC       |
| TT         | MM   | MM   | PP       | DD       | SS       | CC       |
| TT         | MM   | MM   | PP       | DD       | SS       | CC       |
| TT         | MM   | MM   | PP       | DD       | SS       | CC       |
| TT         | MM   | MM   | PP       | DDDDDDDD | SSSSSSSS | CCCCCCCC |
| TT         | MM   | MM   | PP       | DDDDDDDD | SSSSSSSS | CCCCCCCC |

```

LL SSSSSSSS TTTT TTTT
LL SSSSSSSS TTTT TTTT
 SS TT
 SS TT
 SS TT
 SS TT
 SSSSSS TT
 SSSSSS TT
 SS TT
 SS TT
 SS TT
 SS TT
 SSSSSSSS TT
 SSSSSSSS TT

```

TMPDST - PARSE DESTINATION DESC MACRO V05.03b Saturday 29-Jun-85 05:33  
Table of contents

|    |     |                               |
|----|-----|-------------------------------|
| 4- | 57  | MACRO DEFINITIONS             |
| 5- | 127 | LOCAL DATA                    |
| 6- | 156 | TPARS STATE TABLES            |
| 7- | 204 | \$TMPDST - BUILD DST TEMPLATE |
| 7- | 242 | SETTYP - SET DESTINATION TYPE |
| 7- | 243 | SETNAM - SET DST NAME         |
| 7- | 244 | SETPRI - SET PRIORITY         |
| 7- | 245 | SETOBJ - SET OBJECT NUMBER    |
| 7- | 246 | SETTSK - SET TASK NAME        |

.TITLE TMPDTE - PARSE LOCAL DTE DESCRIPTOR  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE LOCAL DTE DESCRIPTOR

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0





| SYMBOL    | VALUE       | REFERENCES        |
|-----------|-------------|-------------------|
| \$EXIT    | = 000000    | #6-107            |
| \$FAIL    | = 177777    | #6-107            |
| \$GPRM    | = *****     | 6-107             |
| \$LAMDA   | = 000000    | #6-107            |
| \$NUMBR   | = 000002    | #6-107            |
| \$QINSN   | = ***** GX  | 7-176             |
| \$RAD50   | = 000016    | #6-107            |
| \$RONLY   | = *****     | 6-107      6-107  |
| \$STRNG   | = 000004    | #6-107            |
| \$SUBXP   | = 000010    | #6-107            |
| \$TIMEVT  | = 000004 RG | #7-158            |
| \$TMLST   | = ***** GX  | 8-194             |
| \$TIALC   | = ***** GX  | 7-161             |
| \$\$\$FLG | = 177777    | #6-107            |
| \$\$\$KEY | = 177777    | #6-107            |
| .PNUMB    | = ***** GX  | 9-225      10-256 |
| .PNUMH    | = ***** GX  | 9-226      10-250 |
| .TPARS    | = ***** GX  | 7-171             |

```

351 .SBTTL Utility subroutines
352 ;+
353 ** MOVSTR - move string
354 ;
355 Inputs: .PSTCN = length
356 .PSTPT = string address
357 R2 = target
358 ;
359 Output: counted string moved to (R2).
360 ;
361 R0, R1, and R2 are clobbered.
362 ;
363 MOVSTR:
364 000500 MOV .PSTCN,R0 ; Get string length
365 000500 MOV .PSTPT,R1 ; Get string address
366 000510 ADD .PSTCN,R1 ; Point past last character
367 000514 CMPB -(R1),#', ; Is last char a comma?
368 000520 BNE 5$; If not, skip
369 000522 DEC R0 ; Else, don't include comma
370 000524 MOVB R0,(R2)+ ; Stuff length
371 000526 MOV .PSTPT,R1 ; reclaim pointer to start
372 000532 MOVB (R1)+(R2)+ ; MOVE...
373 000534 SOB R0,10$; ...IT
374 000536 RETURN
375 .END
376 000001

```

```

193 .SBTTL SETWD1 - SETUP INITIAL FEATURES WORD
194 ;+
195 ; *** - SETWD1 - SETUP INITIAL FEATURES WORD
196 ;
197 ; THIS ACTION ROUTINE SETS UP THE INITIAL FEATURES WORD.
198 ;
199 ; INPUT:
200 ; ,PNUMB = THE INITIAL FEATURES WORD
201 ; ,PNUMH = THE INITIAL FEATURES WORD (HIGH ORDER)
202 ; TEMP = TEMPLATE ADDRESS
203 ;
204 ; OUTPUT:
205 ; C.WD1, $CNFIG = THE INITIAL FEATURES WORD
206 ;
207 ; -
208
209 000106 SETWD1:
210 000106 MOV TEMP,R0 ; GET START ADDRESS OF TEMPLATE
211 000112 016700 177666 MOV ,PNUMB,C.WD1(R0) ; STORE THE INITIAL FEATURES WORD
212 000120 016760 000000G 000006 MOV ,PNUMB,$CNFIG ; STORE IT GLOBALLY, TOO
213 000126 005767 000000G 000000G TST ,PNUMH ; IS THIS A WORD VALUE ?
214 000132 001402 BEQ 10$; IF EQ, YES .. OKAY
215 000134 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
216 000140 000207 10$: RETURN

```

```

205 .SBTTL SETNAM - SET UP PROCESS NAME
206 ;+
207 *** - SETNAM - SETUP PROCESS NAME
208 :
209 THIS ACTION ROUTINE SETS UP THE PROCESS NAME.
210 :
211 INPUT:
212 .PSTPT = START ADDRESS OF THE PROCESS NAME IN ASCII
213 .PSTCN = NUMBER OF CHARACTERS IN THE PROCESS NAME
214 :
215 OUTPUT:
216 C.NAM = RAD50 PROCESS NAME
217 :
218 -
219
220 SETNAM:
221 000074 016700 000000G MOV .PSTPT,R0 ; GET ADDRESS OF ASCII PROCESS NAME
222 000100 010701 000000G MOV PC,R1 ; PERIODS ARE ACCEPTABLE
223 000102 004767 000000G CALL $CAT5 ; CONVERT PROCESS NAME TO RAD50
224 000106 016700 177670 MOV TEMP,R0 ; GET START OF TEMPLATE
225 000112 010160 000004 MOV R1,C.NAM(R0) ; STORE PROCESS NAME IN TEMPLATE
226 000116 022767 000003 000000G CMP #3,.PSTCN ; IS THE PROCESS NAME TOO BIG ?
227 000124 103002 000002 BHIS 10$; IF HIS, NO
228 000126 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
229 000132 000207 10$ RETURN

```

6 10  
TMPLOG - CFE PARSE LOGGING STAT MACRO V05.03b Saturday 29-Jun-85 05:36  
Table of contents

|    |     |                                  |
|----|-----|----------------------------------|
| 4- | 55  | MACRO DEFINITIONS                |
| 5- | 72  | LOCAL DATA                       |
| 6- | 93  | STATE TABLE                      |
| 7- | 113 | \$TMLOG - SETUP LOGGING TEMPLATE |
| 8- | 141 | SETSTA - SET UP LOGGING STATE    |

```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 ;****
65 ; MACRO CALLS
66 ;****
67
68 .MCALL ISTAT$,STAT$,TRANS$
69
70 ;****
71 ; LOCAL DATA
72 ;****
73
74 000074 LA = '<' ; LEFT ANGLE BRACKET
75 000076 RA = '>' ; RIGHT ANGLE BRACKET
76 000000 TEMP: .BLKW 1 ; ADDRESS OF NODE TEMPLATE
77
78 ; DEFINE NODE TEMPLATE
79 ;
80 000000 .ASECT
81 000000 .=0
82 000000 .BLKW 1 ; C.LNK
83 000002 .BLKW 1 ; C.STS
84 000004 C.NAM: .BLKW 1 ; NODE NAME (RAD50)
85 000006 .BLKW 1 ;
86 000010 C.NID: .BLKW 1 ; NODE ID
87 000012 .BLKB 32. ; LENGTH AND STRING
88 000052 C.NUM: .BLKW 1 ; NODE ADDRESS
89 000054 C.HOS: .BLKW 1 ; HOST NODE ADDRESS
90 000056 C.LEN=.
91 000002 .PSECT
92

```

TMPOBJ - CFE PARSE OBJECT DEFIN MACRO V05.03b Saturday 29-Jun-85 05:37 <sup>6.12</sup>  
Table of contents

|     |     |                                                     |
|-----|-----|-----------------------------------------------------|
| 4-  | 55  | MACRO DEFINITIONS                                   |
| 6-  | 146 | \$TMOBJ - SETUP OBJECT TEMPLATE                     |
| 7-  | 181 | FINOBJ - FIND NEXT OBJECT IN SORTED LIST            |
| 8-  | 204 | SETYPE - SETUP OBJECT TYPE                          |
| 9-  | 229 | SETNAM - SET UP OBJECT NAME                         |
| 10- | 260 | SETVfy - SETUP OBJECT VERIFICATION LEVEL            |
| 11- | 287 | SETFLG - SETUP THE FLAGS BYTE                       |
| 12- | 314 | SETCPY - SETUP THE MAXIMUM NUMBER OF COPIES ALLOWED |



IMPOBJ      CREATED BY    MACRO    ON 29-JUN-85 AT 05:37      PAGE 2      F 13

MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |  |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--|
| DBGTP\$ | #5-101 | #5-112 | #5-134 |        |        |        |        |        |        |        |  |
| ISTAT\$ | #4-71  | 5-101  |        |        |        |        |        |        |        |        |  |
| MTRANS  | #5-101 |        |        |        |        |        |        |        |        |        |  |
| PRINT   | #4-62  |        |        |        |        |        |        |        |        |        |  |
| STATE\$ | #4-71  | 5-104  | #5-107 | #5-110 | #5-114 | #5-117 | #5-120 | #5-123 | #5-126 | #5-129 |  |
|         | #5-132 | #5-136 | #5-141 | #5-144 |        |        |        |        |        |        |  |
| TRANS   | #4-71  | #5-105 | #5-108 | #5-111 | #5-112 | #5-115 | #5-118 | #5-121 | #5-124 | #5-127 |  |
|         | #5-130 | #5-133 | #5-134 | #5-137 | #5-142 |        |        |        |        |        |  |

F 14

TMPPAR - CFE PARSE PARTITION DE MACRO V05.03b Saturday 29-Jun-85 05:38 Page 11-1  
Symbol table

|         |         |     |        |          |      |          |         |          |          |          |     |
|---------|---------|-----|--------|----------|------|----------|---------|----------|----------|----------|-----|
| ALL     | 000076R | 002 | FLAG   | 000000R  | TEMP | 000002R  | \$EOS = | 000012   | \$TMPAR  | 000004RG |     |
| ALLC    | 000066R | 002 | NEXT   | 000010R  | 002  | TOP      | 000034R | \$ERROR= | *****    | GX       |     |
| CS.PAR= | *****   | GX  | PARKTB | 000000RG | 003  | UMRS     | 000046R | \$EXIT = | 000000   |          |     |
| CS.TOP= | *****   | GX  | PARSTB | 000000RG | 002  | \$ALPHA= | 000022  | \$FAIL = | 177777   |          |     |
| C.ALL   | 000016  |     | SBCHK  | 000104R  | 002  | \$ANY =  | 000020  | \$LAMDA= | 000000   |          | 004 |
| C.EXT   | 000014  |     | SETALL | 000272R  |      | \$BLANK= | 000006  | \$NUMBR= | 000002   |          |     |
| C.LEN = | 000020  |     | SETEXT | 000244R  |      | \$BYTX7= | *****   | GX       | \$RAD50= | 000016   |     |
| C.NAM0  | 000004  |     | SETFLG | 000356R  |      | \$CAT5 = | *****   | GX       | \$STRNG= | 000004   |     |
| C.NAM1  | 000010  |     | SETNMO | 000104R  |      | \$DIGIT= | 000024  |          | \$SUBXP= | 000010   |     |
| C.STS = | *****   | GX  | SETNMI | 000164R  |      | \$DNUMB= | 000014  |          | \$TALOC= | *****    | GX  |
| EXIT    | 000116R | 002 |        |          |      |          |         |          |          |          |     |

. ABS. 000020 000 (RW,I,GBL,ABS,OVR)  
000372 001 (RW,I,LCL,REL,CON)  
\$STATE 000124 002 (RW,D,LCL,REL,CON)  
\$KTAB 000006 003 (RW,D,LCL,REL,CON)  
\$KSTR 000020 004 (RW,D,LCL,REL,CON)

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10422 Words ( 41 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:29.24

SY:TMPPAR.V2,[132,134]TMPPAR/CR/-SP=SY:[1,1]RSXMCML/ML,[130,110]NETLIB/ML,[132,10]TMPPAR

Symbol table

|                  |                 |                       |                   |                    |
|------------------|-----------------|-----------------------|-------------------|--------------------|
| CS.FLG= ***** GX | ERRPOR 000100R  | SETPOR 000376R        | \$EOS = 000012    | \$SUBXP= ^00010    |
| CS.PSN= ***** GX | ERRPSN 000000R  | SETTHI 001470R        | \$ERROR= ***** GX | \$TALOC= .***** GX |
| CS.THI= ***** GX | ERRRTRN 000156R | SETTLO 000516R        | \$EXIT = 000000   | \$TMPSN 000226RG   |
| CS.TLO= ***** GX | EX\$SEV= 000004 | \$ALPHA= 000022       | \$FAIL = 177777   | \$\$\$FLG= 177777  |
| C.FLG 000012     | PSNK7B 000000RG | 003 \$ANY = 000020    | \$GTNUM= ***** GX | \$\$\$KEY= 177777  |
| C.LEN = 000020   | PSNSTB 000000RG | 002 \$BLANK= 000006   | \$GTR50= ***** GX | \$\$\$STA= 000000  |
| C.NAM 000004     | PSN2 000036R    | 002 \$BYTMN= ***** GX | \$LAMBDA= 000000  | .ERROR= ***** GX   |
| C.POR 000010     | PSN3 000064R    | 002 \$CFERR= ***** GX | \$NUMBR= 000002   | .FLAGS= ***** GX   |
| C.THI 000016     | PSN4 000112R    | 002 \$CLERR= ***** GX | \$PSIPT= ***** GX | .PNUMB= ***** GX   |
| C.TLO 000014     | PSN5 000140R    | 002 \$DIGIT= 000024   | \$RAD50= 000016   | .TEMP = ***** GX   |
| ERRFLG 000127R   | SETFLG 000442R  | \$DNUMB= 000014       | \$STRNG= 000004   | .TPARS= ***** GX   |
| ERRNAM 000042R   | SETNAM 000362R  |                       |                   |                    |

. ABS. 000020 000 (RW,I,GBL,ABS,OVR)  
00054 001 (RW,I,LCL,REL,CON)  
\$STATE 000144 002 (RW,D,LCL,REL,CON)  
\$KTAB 000000 003 (RW,D,LCL,REL,CON)  
\$KSTR 000000 004 (RW,D,LCL,REL,CON)

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 11032 Words ( 44 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:30.54  
SY:TMPPSN.V2,[132,134]TMPPSN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPPSN

```

303 ; ** SETLCN - SET LOGICAL CHANNEL NUMBER
304 ; ** SETCT - SET COUNTER TIMER
305 ; ** SETPSZ - SET MAXIMUM BLOCK SIZE
306 ; ** SETWSZ - SET WINDOW SIZE
307 ; ** SETOWN - SET OWNER PROCESS
308 ; ** SETFLG - SET FLAGS BYTE
309 ; ** SETSTA - SET PVC STATE
310
311 ; INPUTS .TPARS VARIABLES.
312
313 ; OUTPUT VALUE SET IN TEMPLATE.
314 ;
315
316 000534 SETNAM: $GTR50 #P$VNAM,#2
317
318 000550 SETLCN: $GTNUM #P$VLCN,#1,#4095.
319
320 000570 SETCT: $GTNUM #P$VCT
321
322 000610 016702 000000G SETSTA: MOV .TEMP,R2 ; POINT TO TEMPLATE
323 000614 116762 177530 000000G MOV STA,P$VSTA(R2) ; STUFF STATE
324 000622 000207 RETURN
325
326 000624 SETPSZ: $GTNUM #P$VPSZ,#16.,#1024.,#PV$PSZ
327
328 000652 SETWSZ: $GTNUM #P$VWSZ,#1,#127.,#PV$WSZ
329
330 000700 SETOWN: $GTR50 #P$VOWN,#1.,#PV$OWN
331
332 000722 SETFLG: $GTNUM #P$VFLG,,,#CS.FLG
333

```

\*\*FILE\*\*[D\*\*TMPDSA

```

TTTTTTTTTT MM MM PPPPPPP DDDDDDD SSSSSSS AAAAAA
TTTTTTTTTT MM MM PPPPPPP DDDDDDD SSSSSSS AAAAAA
 TT MM MM MM PP DD SS AA AA
 TT MM MM MM PP DD SS AA AA
 TT MM MM MM PP DD SS AA AA
 TT MM MM MM PP DD SS AA AA
 TT MM MM MM PPPPPPP DD DD SSSSSS AA AA
 TT MM MM MM PPPPPPP DD DD SSSSSS AA AA
 TT MM MM MM PP DD DD SS AAAAAAAAAA
 TT MM MM MM PP DD DD SS AAAAAAAAAA
 TT MM MM MM PP DD DD SS AA AA
 TT MM MM MM PP DD DD SS AA AA
 TT MM MM MM PP DDDDDDD SSSSSSS AA AA
 TT MM MM MM PP DDDDDDD SSSSSSS AA AA

```

```

....
....
....
....

```

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLL SS:SSSSS TT
LLLLLLLLLL SSSSSSS TT

```

TMPDSC - PARSE DESTINATION CALL MACRO V05.03b Saturday 29-Jun-85 05:32 <sup>H 2</sup>  
Table of contents

|    |     |                               |
|----|-----|-------------------------------|
| 4- | 57  | MACRO DEFINITIONS             |
| 5- | 126 | LOCAL DATA                    |
| 6- | 138 | TPARS STATE TABLES            |
| 7- | 186 | \$TMPDSC - BUILD DSC TEMPLATE |
| 8- | 228 | SETMSK - SET CALL MASK        |
| 8- | 229 | SETVAL - SET CALL VALUE       |

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55

.TITLE TMPDST - PARSE DESTINATION DESCRIPTOR  
.IDENT /V05.00/  
.ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

# MODULE DESCRIPTION:

STATE TABLE TO PARSE DESTINATION DESCRIPTOR

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

## IDENT HISTORY:

- 1.00 14-AUG-81  
DECNET-11M/S V3.1  
DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/Rsx V1.0

```

57 .SBTTL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61
62 .MACRO PRINT TEXT
63 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM PRINT
66
67 .MACRO ERROR$ TEXT
68
69 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
70 ; MESSAGE STRING.
71
72 .IF DIF <TEXT><R0>
73 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
74 .ENDC
75 CALL $CFERR ; PRINT ERROR MESSAGE
76 .ENDM ERROR$
77
78 .MACRO EPRINT TEXT
79
80 ; PRINT TEXT ON ERROR LUN
81
82 .IF DIF <TEXT><R0>
83 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
84 .ENDC
85 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
86 .ENDM EPRINT
87
88 ; GET NUMBER WITH RANGE CHECKING
89
90 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
91 MOV OFFS,R2
92 MOV LO,R0
93 MOV HI,R1
94 .IF NB <OPT>
95 BIS OPT,@.FLAGS
96 .ENDC
97 JMP $GTNUM
98 .ENDM
99
100 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
101 MOV OFFS,R2
102 MOV LO,R0
103 MOV HI,R1
104 .IF NB <OPT>
105 BIS OPT,@.FLAGS
106 .ENDC
107 JMP $GTNMB
108 .ENDM
109
110 ; GET RAD50 WITH FIELD SIZE CHECKING
111
112 .MACRO $GTR50 OFFS,LEN,OPT
113 MOV OFFS,R2

```



TMPEVT - CFE PARSE EVENT DEFINI MACRO V05.03b Saturday 29-Jun-85 05:34 <sup>H 5</sup>  
Table of contents

|     |     |                                          |
|-----|-----|------------------------------------------|
| 4-  | 55  | MACRO DEFINITIONS                        |
| 5-  | 75  | LOCAL DATA                               |
| 6-  | 100 | STATE TABLE                              |
| 7-  | 145 | \$TIMEVT - SETUP EVENT TEMPLATE          |
| 8-  | 181 | FINEVT - FIND NEXT EVENT IN SORTED LIST  |
| 9-  | 206 | SETCLS - SETUP EVENT CLASS               |
| 10- | 234 | SETMSK - SET UP EVENT MASKS              |
| 11- | 262 | SETMON - SETUP MONITOR LOGGING SINK TYPE |
| 11- | 263 | SETCON - SETUP CONSOLE LOGGING SINK TYPE |
| 11- | 264 | SETFIL - SETUP FILE LOGGING SINK TYPE    |

TMPEVT      CREATED BY    MACRO    ON 29-JUN-85 AT 05:34      PAGE 3      G 6

MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #6-107     | #6-127 | #6-128 | #6-141 |        |        |        |        |        |        |
| FLTDF\$    | #4-70      | 4-72   |        |        |        |        |        |        |        |        |
| ISTAT\$    | #4-70      | 6-107  |        |        |        |        |        |        |        |        |
| MTRAN\$    | #6-107     |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-61      |        |        |        |        |        |        |        |        |        |
| STAT\$     | #4-70      | 6-110  | #6-113 | #6-116 | #6-119 | #6-122 | #6-125 | #6-130 | #6-136 | #6-139 |
|            | #6-143     |        |        |        |        |        |        |        |        |        |
| TRANS      | #4-70      | #6-111 | #6-114 | #6-117 | #6-120 | #6-123 | #6-126 | #6-127 | #6-128 | #6-131 |
|            | #6-137     | #6-140 | #6-141 |        |        |        |        |        |        |        |

6 7

TMPFIL - CFE PARSE REMOTE CHARA MACRO V05.03b Saturday 29-Jun-85 05:34 Page 8-1  
 Symbol table

|                  |                    |                       |                   |                        |
|------------------|--------------------|-----------------------|-------------------|------------------------|
| CKFSP 000254R    | F\$DIA 000012      | MOVSTR 000500R        | \$CFERR= ***** GX | \$SUBXP= 000010        |
| CKUIC 000310R    | F\$DUM 000014      | MXFSP = 000036        | \$DIGIT= 000024   | \$TALOC= ***** GX      |
| CLEFIL 000232R   | F\$LD 000010       | REJ 000274R           | \$DNUMB= 000014   | \$TMFIL 000124RG       |
| COMMA = 000054   | F\$LEN = 000016    | STFIL 000336R         | \$EOS = 000012    | \$T1ALC= ***** GX      |
| CS.FIL= ***** GX | F\$SEC 000004      | STRING 000166R        | \$ERROR= ***** GX | \$\$\$FLG= 177777      |
| ERRFSP 000045R   | F\$TER 000006      | S10 000026R           | \$EXIT = 000000   | \$\$\$KEY= 177777      |
| ERRUIC 000077R   | F.CNT 000002       | S20 000054R           | \$FAIL = 177777   | \$\$\$STA= 000224R 002 |
| EX\$SEV= 000004  | F.DEV 000005       | S30 000102R           | \$FILHD= ***** GX | .ERROR= ***** GX       |
| FD 000002R       | F.LEN = 000043     | S40 000130R           | \$LAMBDA= 000000  | .FLAGS= ***** GX       |
| FILE 000000R     | F.LNK 000000       | \$ALPHA= 000022       | \$NUMBR= 000002   | .PSTCN= ***** GX       |
| FILKTR 000000RG  | 003 GETUIC 000214R | 002 \$ANY = 000020    | \$RAD50= 000016   | .PSTPT= ***** GX       |
| FILSPC 000154R   | 002 GTRST 000200R  | 002 \$BLANK= 000006   | \$SAVRG= ***** GX | .TEMP = ***** GX       |
| FILSTB 000000RG  | 002 GTUIC 000224R  | 002 \$BYTMN= ***** GX | \$STRNG= 000004   | .TPARS= ***** GX       |

. ABS. 000043 000 (RW,I,GBL,ABS,OVR)  
 000540 001 (RW,I,LCL,REL,CON)  
 \$STATE 000240 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000000 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000000 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10840 Words ( 43 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operatino system: RSX-11M/PLUS

Elapsed time: 00:00:38.04  
 SY:TMPFIL.V2,[132,134]TMPFIL/CR/-SP=SY:[1,1]RSXMCML.SML/ML,[130,110]NETLIB/ML,[132,10]TMPFIL

```

218 .SBTTL SETWD2 - SETUP THE SETTABLE STATES WORD
219 ;+
220 ; *** - SETWD2 - SET THE SETTABLE STATES WORD
221 ;
222 ; INPUT:
223 ; .PNUMB = THE SETTABLE STATES WORD
224 ; .PNUMH = THE SETTABLE STATES WORD (HIGH ORDER)
225 ; TEMP = THE TEMPLATE ADDRESS
226 ;
227 ; OUTPUT:
228 ; NUMBER IS STORED IN C.WD2 OF TEMPLATE
229 ;
230 ; -
231
232 000142 SETWD2:
233 000142 016700 177632 MOV TEMP,R0 ; POINT R0 AT TEMPLATE
234 000146 016760 000000G 000010 MOV .PNUMB,C.WD2(R0) ; STORE THE THE SETTABLE STATES WORD
235 000154 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
236 000160 001402 BEQ 10$; IF EQ, YES .. OKAY
237 000162 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
238 000166 000207 10$: RETURN

```

```

231
232
233
234
235
236
237
238
239
240
241
242
243
244 000134
245 000134 016700 177642
246 000140 016760 177634 000006
247 000146 032767 000000G 177624
248 000154 001002
249 000156 062716 000002
250 000162 000207

.SBTTL SETFLG - SETUP THE FLAGS WORD
:
: +
: *** - SETFLG - SETUP THE FLAGS WORD
:
: INPUT:
: FLAG = THE FLAGS WORD
: TEMP = START ADDRESS OF THE TEMPLATE
:
: OUTPUT:
: THE FLAGS WORD IS MOVED INTO THE TEMPLATE
:
: -
:
SETFLG:
MOV TEMP,R0 ; GET START OF TEMPLATE
MOV FLAG,C.FLG(R0) ; STORE THE FLAGS WORD
BIT #ZF.LLC,FLAG ; ZF.LLC MUST BE SET
BNE 10$; IF NE, IT IS
ADD #2,(SP) ; ELSE REJECT TRANSITION
10$: RETURN

```

.TITLE TMPLOG - CFE PARSE LOGGING STATE DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE OBJECT DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 02-JUL-80  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/Rsx V1.0

```

94 ;****
95 ; INITIALIZE TABLE GENERATION
96 ;****
97
98
99 000002 ISTAT$ NODSTB,NODKTB
100
101
102 000002 STATES$
103 000002 TRANS$ LA
104
105 000002 STATES$
106 000002 TRANS$ $RAD50,,SETNAM
107
108 000002 STATES$
109 000002 TRANS$ RA
110
111 000002 STATES$
112 000002 TRANS$ <','>
113
114 000002 STATES$
115 000002 TRANS$ LA
116
117 000002 STATES$
118 000002 TRANS$!ID,,SETNID
119
120 000002 STATES$
121 000002 TRANS$ <','>
122
123 000002 STATES$
124 000002 TRANS$ $NUMBR,,SETNNA
125
126 000002 STATES$ NN2
127 000002 TRANS$ <','>
128
129 000002 STATES$
130 000002 TRANS$ $NUMBR,,SETNNM
131
132 000002 STATES$
133 000002 TRANS$ <','>
134
135 000002 STATES$
136 000002 TRANS$ $NUMBR,,SETHOA
137
138 000002 STATES$
139 000002 TRANS$ <','>
140
141 000002 STATES$
142 000002 TRANS$ $NUMBR,$EXIT,SETHOS
143
144 000002 STATES$ ID
145 000002 TRANS$ RA,$EXIT
146 000002 TRANS$ $ANY,ID
147
148 000002 STATES$

```

.TITLE TMPOBJ - CFE PARSE OBJECT DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE OBJECT DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RX V1.0





TMPPAR CREATED BY MACRO ON 29-JUN-85 AT 05:38 PAGE 1 G 14  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL    | VALUE      | REFERENCES                                               |
|-----------|------------|----------------------------------------------------------|
| CS.PAR    | = ***** GX | 6-178                                                    |
| C.ALL     | 000016     | #4-88 *6-179 *10-295 10-302                              |
| C.EXT     | 000014     | #4-87 *9-271                                             |
| C.LEN     | = 000020   | #4-89 6-169                                              |
| C.NAMO    | 000004     | #4-83 6-173 *7-210 *7-215                                |
| C.NAM1    | 000010     | #4-85 *8-242 *8-247                                      |
| C.STS     | = ***** GX | *6-178 *11-323                                           |
| FLAG      | 000000 R   | #4-74 11-323                                             |
| PARKTB    | 000000 RG  | #5-97 6-181                                              |
| PARSTB    | 000000 RG  | #5-97 6-182                                              |
| SETALL    | 000272 R   | #10-293                                                  |
| SETEXT    | 000244 R   | #9-269                                                   |
| SETFLG    | 000356 R   | #11-321                                                  |
| SETNMO    | 000104 R   | #7-205                                                   |
| SETNM1    | 000164 R   | #8-237                                                   |
| TEMP      | 000002 R   | #4-75 *6-171 7-209 7-214 8-241 8-246 9-270 10-294 11-322 |
| \$ALPHA   | = 000022   | #5-97                                                    |
| \$ANY     | = 000020   | #5-97                                                    |
| \$BLANK   | = 000006   | #5-97                                                    |
| \$BYTXT   | = ***** GX | *10-306                                                  |
| \$CAT5    | = ***** GX | 7-208 7-213 8-240 8-245                                  |
| \$DIGIT   | = 000024   | #5-97                                                    |
| \$DNUMB   | = 000014   | #5-97                                                    |
| \$EOS     | = 000012   | #5-97                                                    |
| \$ERROR   | = ***** GX | *6-185                                                   |
| \$EXIT    | = 000000   | #5-97                                                    |
| \$FAIL    | = 177777   | #5-97                                                    |
| \$GPRM    | = *****    | 5-97                                                     |
| \$LAMDA   | = 000000   | #5-97                                                    |
| \$NUMBR   | = 000002   | #5-97                                                    |
| \$RAD50   | = 000016   | #5-97                                                    |
| \$RONLY   | = *****    | 5-97 5-97                                                |
| \$STRNG   | = 000004   | #5-97                                                    |
| \$SUBXP   | = 000010   | #5-97                                                    |
| \$TALOC   | = ***** GX | 6-170                                                    |
| \$TMPPAR  | 000004 RG  | #6-167                                                   |
| \$\$\$FLG | = 177777   | #5-97                                                    |
| \$\$\$KEY | = 177777   | #5-97                                                    |
| .PNUMB    | = ***** GX | 9-271 10-295 10-298                                      |
| .PNUMH    | = ***** GX | 9-272 10-296                                             |
| .PSTCN    | = ***** GX | 7-216 8-248                                              |
| .PSTPT    | = ***** GX | 7-206 8-238                                              |
| .TPARS    | = ***** GX | 6-183                                                    |

TMPPSN CREATED BY MACRO ON 29-JUN-85 AT 05:38 PAGE 1 G 15  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL    | VALUE       | REFERENCES                           |
|-----------|-------------|--------------------------------------|
| CS.FLG    | = ***** GX  | 7-289                                |
| CS.PSN    | = ***** GX  | 7-241                                |
| CS.THI    | = ***** GX  | 7-291                                |
| CS.TLO    | = ***** GX  | 7-293                                |
| C.FLG     | = 000012    | #5-136 7-289                         |
| C.LEN     | = 000020    | #5-139 7-235                         |
| C.NAM     | = 000004    | #5-134 7-279                         |
| C.POR     | = 000010    | #5-135 7-287                         |
| C.THI     | = 000016    | #5-138 7-291                         |
| C.TLO     | = 000014    | #5-137 7-293                         |
| ERRFLG    | 000127 R    | #5-149                               |
| ERRNAM    | 000042 R    | #5-147                               |
| ERRPOR    | 000100 R    | #5-148                               |
| ERRPSN    | 000000 R    | #5-146 7-256                         |
| ERRTRN    | 000156 R    | #5-150                               |
| EX\$SEV   | = 000004    | #5-144 5-146 5-147 5-148 5-149 5-150 |
| PSNKTB    | = 000000 RG | #6-159 7-244                         |
| PSNSTB    | = 000000 RG | #6-159 7-245                         |
| SETFLG    | 000442 R    | #7-289                               |
| SETNAM    | 000362 R    | #7-279                               |
| SETPOR    | 000376 R    | #7-281                               |
| SETTHI    | 000470 R    | #7-291                               |
| SETTLO    | 000516 R    | #7-293                               |
| \$ALPHA   | = 000022    | #6-159                               |
| \$ANY     | = 000020    | #6-159                               |
| \$BLANK   | = 000006    | #6-159                               |
| \$BYTMN   | = ***** GX  | *7-286                               |
| \$CFERR   | = ***** GX  | 7-250 7-256                          |
| \$DIGIT   | = 000024    | #6-159                               |
| \$DNUMB   | = 000014    | #6-159                               |
| \$EOS     | = 000012    | #6-159                               |
| \$ERROR   | = ***** GX  | *7-251 *7-252 *7-257                 |
| \$EXIT    | = 000000    | #6-159                               |
| \$FAIL    | = 177777    | #6-159                               |
| \$GPRM    | = *****     | 6-159                                |
| \$GTNUM   | = ***** GX  | 7-287 7-289 7-291 7-293              |
| \$GTR50   | = ***** GX  | 7-279                                |
| \$LAMDA   | = 000000    | #6-159                               |
| \$NUMBR   | = 000002    | #6-159                               |
| \$PSIPT   | = ***** GX  | 7-232 *7-238                         |
| \$RAD50   | = 000016    | #6-159                               |
| \$RONLY   | = *****     | 6-159 6-159                          |
| \$STRNG   | = 000004    | #6-159                               |
| \$SUBXP   | = 000010    | #6-159                               |
| \$TALOC   | = ***** GX  | 7-236                                |
| \$TMPSN   | = 000226 RG | #7-231                               |
| \$\$\$FLG | = 177777    | #6-159                               |
| \$\$\$KEY | = 177777    | #6-159                               |
| .ERROR    | = ***** GX  | *7-242 7-248 7-291 7-293             |
| .FLAGS    | = ***** GX  | *7-240 7-289                         |
| .PNUMB    | = ***** GX  | 7-281                                |
| .TEMP     | = ***** GX  | *7-237                               |

SETDEV - SET DEVICE NAME IN LINE ID

```

335 .SBTTL SETDEV - SET DEVICE NAME IN LINE ID
336 .SBTTL SETCON - SET CONTROLLER NUMBER IN LINE ID
337 .SBTTL SETTRB - SET TRIB NUMBER IN LINE ID
338
339 ;+
340 ; ** SETDEV - SET DEVICE NAME IN LINE ID
341 ; ** SETCON - SET CONTROLLER NUMBER IN LINE ID
342 ; ** SETTRB - SET TRIB NUMBER IN LINE ID
343 ;
344 ; INPUTS: .TPARS CONVENTIONS.
345 ; OUTPUT: VALUE SET IN TEMPLATE.
346 ; -
347
348 000750 SETDEV: $GTR50 #P$VNAM,#1
349
350 000764 SETCON: $GTNMB #P$VNAM+2
351
352 001004 SETTRB: $GTNMB #P$VNAM+3
353
354
355
356 000001 .END

```

TMPDSA - PARSE DESTINATION ADDR MACRO V05.03b Saturday 29-Jun-85 <sup>H 1</sup> 05:32  
Table of contents

|    |     |                                    |
|----|-----|------------------------------------|
| 4- | 57  | MACRO DEFINITIONS                  |
| 5- | 126 | LOCAL DATA                         |
| 6- | 140 | TPARS STATE TABLES                 |
| 7- | 196 | \$TMDSA - BUILD DSA TEMPLATE       |
| 8- | 233 | SETNUM - SET REMOTE DTE NUMBER     |
| 8- | 234 | SETSLO - SET SUBADDRESS LOW RANGE  |
| 8- | 235 | SETSHI - SET SUBADDRESS HIGH RANGE |
| 8- | 236 | SETCUG - SET USER GROUP NAME       |

.TITLE TMPDSC - PARSE DESTINATION CALL PARAMETERS  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE DESTINATION CALL PARAMETERS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RX V1.0

```

57 .SBTIL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61 .MACRO PRINT TEXT
62 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
63 CALL $PRINT ; PRINT MESSAGE
64 .ENDM PRINT
65
66 .MACRO ERROR$ TEXT
67
68 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
69 ; MESSAGE STRING.
70
71 .IF DIF <TEXT><R0>
72 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
73 .ENDC
74 CALL $CFERR ; PRINT ERROR MESSAGE
75 .ENDM ERROR$
76
77 .MACRO EPRINT TEXT
78
79 ; PRINT TEXT ON ERROR LUN
80
81 .IF DIF <TEXT><R0>
82 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
83 .ENDC
84 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
85 .ENDM EPRINT
86
87 ; GET NUMBER WITH RANGE CHECKING
88
89 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
90 MOV OFFS,R2
91 MOV LO,R0
92 MOV HI,R1
93 .IF NB <OPT>
94 BIS OPT,@.FLAGS
95 .ENDC
96 JMP $GTNUM
97 .ENDM
98
99 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
100 MOV OFFS,R2
101 MOV LO,R0
102 MOV HI,R1
103 .IF NB <OPT>
104 BIS OPT,@.FLAGS
105 .ENDC
106 JMP $GTNMB
107 .ENDM
108
109 ; GET RAD50 WITH FIELD SIZE CHECKING
110
111 .MACRO $GTR50 OFFS,LEN,OPT
112 MOV OFFS,R2
113

```

```
114 MOV LEN,R1
115 .IF NB <OPT>
116 BIS OPT,@.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 ;
122 ; MACRO CALLS
123 ;
124 .MCALL ISTAT$,STATE$,TRANS$,CALLR
125
```



1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53

.TITLE TMPEVT - CFE PARSE EVENT DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE OBJECT DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 02-JUL-80  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

\*\*FILE\*\*ID\*\*TMPFIL

```

TTTTTTTTT MM MM PPPPPPP FFFFFFFF I11111 LL
TTTTTTTTT MM MM PPPPPPP FFFFFFFF I11111 LL
TT MMMM MM PP PP FF I1 LL
TT MMMM MM PP PP FF I1 LL
TT MM MM MM PP PP FF I1 LL
TT MM MM MM PPPPPPP FFFFFFFF I1 LL
TT MM MM PPPPPPP FFFFFFFF I1 LL
TT MM MM PP FF I1 LL
TT MM MM PP FF I1 LL
TT MM MM PP FF I1 LL
TT MM MM PP FF I1 LL
TT MM MM PP FF I11111 LLLLLLLLLL
TT MM MM PP FF I11111 LLLLLLLLLL

```

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSS TT
LL SSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSS TT
LLLLLLLLL SSSSSSS TT
LLLLLLLLL SSSSSSS TT

```

TMPFIL CREATED BY MACRO ON 29-JUN-85 AT 05:35 PAGE 1 H 7  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL    | VALUE      | REFERENCES                                             |
|-----------|------------|--------------------------------------------------------|
| CKFSP     | 000254 R   | #7-258                                                 |
| CKUIC     | 000310 R   | #7-278                                                 |
| CLEFIL    | 000232 R   | #7-242                                                 |
| COMMA     | = 000054   | #5-130                                                 |
| CS.FIL    | = ***** GX | 6-217                                                  |
| ERRFSP    | 000045 R   | #4-122 7-264                                           |
| ERRUIC    | 000077 R   | #4-123 7-281                                           |
| EX\$SEV   | = 000004   | #4-121 4-122                                           |
| FD        | 000002 R   | #4-115 7-244 7-261 7-309 7-330 7-337                   |
| FILE      | 000000 R   | #4-114 *7-243 7-345                                    |
| FILKTB    | 000000 RG  | #5-132 6-220                                           |
| FILSTB    | 000000 RG  | #5-132 6-221                                           |
| FSDIA     | 000012     | #4-92                                                  |
| FSDUM     | 000014     | #4-93                                                  |
| FSLD      | 000010     | #4-91                                                  |
| F\$LEN    | = 000016   | #4-94 6-212                                            |
| F\$SEC    | 000004     | #4-89                                                  |
| F\$TER    | 000006     | #4-90                                                  |
| F.CNT     | 000002     | #4-106 *7-318                                          |
| F.DEV     | 000005     | #4-108 7-261 7-308 7-309 7-310 7-310 7-329 7-330 7-337 |
| F.LEN     | = 000043   | #4-109 4-115 7-245 7-310 7-323 7-329                   |
| F.LNK     | 000000     | #4-105                                                 |
| MOVSTR    | 000500 R   | 7-262 #8-363                                           |
| MXFSP     | = 000036   | #4-101 4-108 7-259                                     |
| REJ       | 000274 R   | 7-260 #7-264                                           |
| STFIL     | 000336 R   | #7-299                                                 |
| \$ALPHA   | = 000022   | #5-132                                                 |
| \$ANY     | = 000020   | #5-132                                                 |
| \$BLANK   | = 000006   | #5-132                                                 |
| \$BYTMN   | = ***** GX | *7-340                                                 |
| \$CFERR   | = ***** GX | 6-226                                                  |
| \$DIGIT   | = 000024   | #5-132                                                 |
| \$DNUMB   | = 000014   | #5-132                                                 |
| \$EOS     | = 000012   | #5-132                                                 |
| \$ERROR   | = ***** GX | *6-227                                                 |
| \$EXIT    | = 000000   | #5-132                                                 |
| \$FAIL    | = 177777   | #5-132                                                 |
| \$FILHD   | = ***** GX | 7-300                                                  |
| \$GPRM    | = *****    | 5-132                                                  |
| \$LAMDA   | = 000000   | #5-132                                                 |
| \$NUMBR   | = 000002   | #5-132                                                 |
| \$RAD50   | = 000016   | #5-132                                                 |
| \$RONLY   | = *****    | 5-132 5-132 5-132                                      |
| \$SAVRG   | = ***** GX | 7-299                                                  |
| \$STRNG   | = 000004   | #5-132                                                 |
| \$SUBXP   | = 000010   | #5-132                                                 |
| \$TALOC   | = ***** GX | 6-213                                                  |
| \$TMFIL   | 000124 RG  | #6-210                                                 |
| \$TIALC   | = ***** GX | 7-324                                                  |
| \$\$\$FLG | = 177777   | #5-132                                                 |
| \$\$\$KEY | = 177777   | #5-132                                                 |
| .ERROR    | = ***** GX | *6-218 6-224 *7-264 *7-281                             |

```

240
241
242
243
244
245
246
247
248
249
250
251
252
253
254 000170
255 000170 016700 177604
256 000174 016760 000000G 000012
257 000202 005767 000000G
258 000206 001402
259 000210 062716 000002
260 000214 000207
261
262
263 000001

.SBTTL SETWD3 - SETUP THE CURRENT STATES WORD
;+
*** - SETWD3 - SETUP THE CURRENT STATES WORD
;
INPUT:
.PNUMB = THE CURRENT STATES WORD
.PNUMH = THE CURRENT STATES WORD (HIGH ORDER)
TEMP = THE TEMPLATE ADDRESS
;
OUTPUT:
C.WD3 = THE CURRENT STATES WORD
;-

SETWD3: MOV TEMP,R0 ; POINT AT TEMPLATE
MOV .PNUMB,C.WD3(R0) ; STORE THE CURRENT STATES
TST .PNUMH ; IS IT A WORD VALUE ?
BEQ 10$; IF EQ, YES.. OKAY
ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
10$: RETURN

.END

```

```

252 .SBTTL SETPRI - SET THE PRIORITY
253 +
254 *** - SETPRI - SET THE PROCESS PRIORITY
255 :
256 THIS ACTION ROUTINE CHECKS THE LLC PRIORITY TO MAKE SURE IT IS VALID
257 AND STORES IT IN THE TEMPLATE.
258 :
259 INPUT:
260 .PNUMB = PRIORITY
261 .PNUMH = PRIORITY (HIGH ORDER)
262 :
263 OUTPUT:
264 THE PRIORITY IS STORED IN C.PRI OF THE TEMPLATE.
265 :
266 -
267
268 000164 SETPRI:
269 000164 016700 177612 MOV TEMP,R0 ; GET START OF TEMPLATE
270 000170 116760 000000G 000010 MOVSB .PNUMB,C.PRI(R0) ; STORE THE PRIORITY IN THE TEMPLATE
271 000176 005767 000000G TST .PNUMB ; CHECK THE PRIORITY
272 000202 001406 BEQ 10$; IF EQ, OKAY FOR RSX-11M/c
273 000204 026727 000000G 000003 CMP .PNUMB,#3 ; RSX-11D/IAS PRIORITY?
274 000212 001402 BEQ 10$; IF EQ, YES - OKAY
275 000214 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
276 000220 000207 10$: RETURN

```

```
55 .SBTTL MACRO DEFINITIONS
56 ;****
57 ; LOCAL MACROS
58 ;****
59
60 .MACRO PRINT TEXT
61 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
62 CALL $PRINT ; PRINT MESSAGE
63 .ENDM PRINT
64
65 ;****
66 ; MACRO CALLS
67 ;****
68
69 .MCALL ISTAT$,STATE$,TRANS$
70
```

```

150 .SBTTL $TMNOD - SETUP NODE TEMPLATE
151 ;+
152 *** - $TMNOD - SETUP NODE TEMPLATE
153
154 INPUT:
155 R3-R5 - TPARS REGISTERS
156
157 OUTPUT:
158 C-BIT = SUCCESS/FAILURE
159 IF SUCCESS, THE NODE TEMPLATE IS STORED IN THE END OF TASK BUFFER.
160
161 ; -
162
163 $TMNOD::
164 000002 010546 MOV R5,-(SP) ; SAVE R5
165 000004 012701 MOV #C.LEN,R1 ; GET LENGTH OF ALLOCATION
166 000010 004767 CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
167 000014 010067 MOV R0,TEMP ; STORE TEMPLATE ADDRESS
168 000020 012767 MOV #1*1024,,$NTADD ; INITIALIZE NODE
169 000026 012760 MOV #1*1024,,$C.NUM(R0) ; ADDRESSES
170 000034 012760 MOV #1*1024,,$C.HOS(R0) ;
171 000042 012760 MOV #CS.NOD,,$STS(R0) ; INDICATE THAT THIS IS A NOD$DF TEMPLATE
172 000050 005001 CLR R1 ; IGNORE BLANKS
173 000052 012702 MOV #NODKTB,R2 ; KEYWORD TABLE
174 000056 012705 MOV #NODSTB,R5 ; STATE TABLE
175 000062 004767 CALL $TPARS ; PARSE THE REST OF THE LINE
176 000066 103002 BCC 20$; NORMAL RETURN IF NO ERROR
177 000070 005267 INC $ERROR ; INDICATE SYNTAX ERROR
178 000074 012605 MOV (SP)+,R5 ; RESTORE R5
179 000076 000207 20$: RETURN

```

```

55 .SBTTL MACRO DEFINITIONS
56
57
58 ;****
59 ; LOCAL MACROS
60 ;****
61
62 .MACRO PRINT TEXT
63 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM PRINT
66
67 ;****
68 ; MACRO CALLS
69 ;****
70
71 .MCALL ISTAT$,STAT$,TRANS$
72
73 ;****
74 ; LOCAL DATA
75 ;****
76
77 LA = '< ; LEFT ANGLE BRACKET
78 RA = '> ; RIGHT ANGLE BRACKET
79 000000 TEMP: .BLKW 1 ; ADDRESS OF NODE TEMPLATE
80
81 ; DEFINE NODE TEMPLATE
82 ;
83 000000 .ASECT
84 000000 .=0
85 000000 .BLKW 1 ; C.LNK
86 000002 .BLKW 1 ; C.STS
87 000004 C.TYP: .BLKB 1 ; OBJECT TYPE
88 000005 C.FLG: .BLKB 1 ; OBJECT FLAGS BYTE
89 000006 C.NAM: .BLKW 2 ; OBJECT TASK NAME (RAD50)
90 000012 C.CPY: .BLKB 1 ; MAXIMUM NUMBER OF COPIES OF A TASK TO SPAWN
91 ; LIMIT IS 63.
92 C.LEN=.
93 000002 .PSECT
94

```



TMPPAR - CFE PARSE PARTITION DE MACRO V05.03b Saturday 29-Jun-85 05:38 <sup>H 13</sup>  
Table of contents

|     |     |                                      |
|-----|-----|--------------------------------------|
| 6-  | 154 | \$TMPAR - SETUP PARTITION TEMPLATE   |
| 7-  | 189 | SETNMO - SET UP PARTITION NAME 0     |
| 8-  | 221 | SETNM1 - SET UP PARTITION NAME 1     |
| 9-  | 223 | SETEXT - CHECK THE EXTENSION         |
| 10- | 277 | SETALL - SET THE BYTE-AREA EXTENSION |
| 11- | 309 | SETFLG - SETUP THE FLAGS WORD        |

TMPPAR      CREATED BY    MACRO    ON 29-JUN-85 AT 05:38      PAGE 2      H 14  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-97      | #5-101 | #5-117 | #5-121 | #5-125 | #5-129 | #5-130 | #5-134 | #5-141 |        |
| ISTAT\$    | #4-68      | 5-97   |        |        |        |        |        |        |        |        |
| MIRAN\$    | #5-97      |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-59      |        |        |        |        |        |        |        |        |        |
| STATE\$    | #4-68      | 5-99   | #5-103 | #5-106 | #5-109 | #5-112 | #5-115 | #5-119 | #5-123 | #5-127 |
|            | #5-132     | #5-136 | #5-139 | #5-143 | #5-149 | #5-152 |        |        |        |        |
| TRANS      | #4-68      | #5-100 | #5-101 | #5-104 | #5-107 | #5-110 | #5-113 | #5-116 | #5-117 | #5-120 |
|            | #5-121     | #5-124 | #5-125 | #5-128 | #5-129 | #5-130 | #5-133 | #5-134 | #5-137 | #5-140 |
|            | #5-141     | #5-144 | #5-150 |        |        |        |        |        |        |        |

TMPPSN      CREATED BY    MACRO    ON 29-JUN-85 AT 05:38      PAGE 2      H 15  
SYMBOL CROSS REFERENCE      CREF    04.00  
SYMBOL    VALUE      REFERENCES  
.TPARS    =    \*\*\*\*\*    GX      7-246

Symbol table

|                   |                       |                       |                   |                    |
|-------------------|-----------------------|-----------------------|-------------------|--------------------|
| CS.FLG= ***** GX  | PVC0 000026R          | 002 P\$VSTA= ***** GX | \$ALPHA= 000022   | \$LAMDA= 000000    |
| CS.PVC= ***** GX  | PVC1 000110R          | 002 P\$VWSZ= ***** GX | \$ANY = 000020    | \$NUMBR= 000002    |
| C\$LENX= ***** GX | PVC2 000142R          | 002 SETCON 000764R    | \$BLANK= 000006   | \$RAD50= 000016    |
| DLM 000352R       | PVC3 000170R          | 002 SETCT 000570R     | \$BYTMN= ***** GX | \$STRNG= 000004    |
| OLMPVC 000244R    | 002 PVC4 000216R      | 002 SETDEV 000750R    | \$CFERR= ***** GX | \$SUBXP= 000010    |
| ERRCT 000100R     | PV\$OWN= ***** GX     | SETFLG 000722R        | \$CLERR= ***** GX | \$TALOC= ***** GX  |
| ERRFLG 000273R    | PV\$PSZ= ***** GX     | SETLCN 000550R        | \$DIGIT= 000024   | \$TMPVC 000354RG   |
| ERRLCN 000031R    | PV\$WSZ= ***** GX     | SETNAM 000534R        | \$DNUMB= 000014   | \$\$\$FLG= 177777  |
| ERRNAM 000000R    | P\$VCT = ***** GX     | SETOWN 000700R        | \$EOS = 000012    | \$\$\$KEY= 000002  |
| ERROWN 000235R    | P\$VFLG= ***** GX     | SETPSZ 000624R        | \$ERROR= ***** GX | \$\$\$STA= 000000  |
| ERRPSZ 000137R    | P\$VLCN= ***** GX     | SETSTA 000610R        | \$EXIT = 000000   | \$\$\$TMP= 000007R |
| ERRSTA 000322R    | P\$VLEN= ***** GX     | SETTRB 001004R        | \$FAIL = 177777   | .ERROR= ***** GX   |
| ERRWSZ 000201R    | P\$VNAM= ***** GX     | SETWSZ 000652R        | \$GTNMB= ***** GX | .FLAGS= ***** GX   |
| EX\$SEV= 000004   | P\$VOWN= ***** GX     | STA 000350R           | \$GTNUM= ***** GX | .TEMP = ***** GX   |
| PVCKTB 000000RG   | 003 P\$VPSZ= ***** GX | X\$LENX= ***** GX     | \$GTR50= ***** GX | .TPARS= ***** GX   |
| PVCSTB 000000RG   | 002                   |                       |                   |                    |

. ABS. 000000 000 (RW,I,GBL,ABS,OVR)  
001024 001 (RW,I,LCL,REL,CON)  
\$STATE 000266 002 (RW,D,LCL,REL,CON)  
\$KTAB 000006 003 (RW,D,LCL,REL,CON)  
\$KSTR 000013 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 11048 Words ( 44 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:48.29  
SY:TMPPVC.V2,[132,134]TMPPVC/CR/-SP=SY:[1,1]RSXMCN.SML/ML,[130,110]NETLIB/ML,[132,10]TMPPVC

.TITLE TMPDSA - PARSE DESTINATION ADDRESS  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE DESTINATION ADDRESS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0

```

57 .SBTTL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61
62 .MACRO PRINT TEXT
63 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM PRINT
66
67 .MACRO ERRORS$ TEXT
68
69 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
70 ; MESSAGE STRING.
71
72 .IF DIF <TEXT><R0>
73 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
74 .ENDC
75 CALL $CFERR ; PRINT ERROR MESSAGE
76 .ENDM ERRORS$
77
78 .MACRO EPRINT TEXT
79
80 ; PRINT TEXT ON ERROR LUN
81
82 .IF DIF <TEXT><R0>
83 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
84 .ENDC
85 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
86 .ENDM EPRINT
87
88 ; GET NUMBER WITH RANGE CHECKING
89
90 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
91 MOV OFFS,R2
92 MOV LO,R0
93 MOV HI,R1
94 .IF NB <OPT>
95 BIS OPT,@.FLAGS
96 .ENDC
97 JMP $GTNUM
98 .ENDM
99
100 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
101 MOV OFFS,R2
102 MOV LO,R0
103 MOV HI,R1
104 .IF NB <OPT>
105 BIS OPT,@.FLAGS
106 .ENDC
107 JMP $GTNMB
108 .ENDM
109
110 ; GET RAD50 WITH FIELD SIZE CHECKING
111
112 .MACRO $GTR50 OFFS,LEN,OPT
113 MOV OFFS,R2

```

```

114 MOV LEN,R1
115 .IF NB <OPT>
116 BIS OPT,@.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 :
122 : MACRO CALLS
123 :
124 .MCALL ISTAT$,STAT$,TRAN$,CALLR
125

```

```

LOCAL DATA
127 .SBTTL LOCAL DATA
128
129 ; DEFINE THE LOCAL DTE PARAMETER BLOCK
130 ;
131 .ASECT
132 .=0
133 .BLKW 1 ; C.LNK
134 .BLKW 1 ; C.STS
135 C.ADR: .BLKW 5. ; RAD50 STRING OF 1.-15. DECIMAL DIGITS
136 C.LIN: .BLKW 1 ; LINE ID: RAD50 PROCESS NAME
137 .BLKW 3 ; CONTROLLER, UNIT, TRIB NUMBERS
138 C.STA: .BLKW 1 ; =0 --> OFF, >0 --> ON
139 C.HSH: .BLKW 1 ; HASH TABLE SIZE (32.-512. ENTRIES)
140 C.CT: .BLKW 1 ; COUNTER TIMER VALUE (0.-65535.)
141 C.NET: .BLKW 2 ; RAD50 NETWORK NAME
142 C.LEN=.
143 .PSECT
144 .NLIST BEX
145
146 ; SYNTAX ERROR DETAIL STRINGS
147 ;
148 EX$SEV = 4
149 000000 004 103 106 ERRADR: .ASCIZ <EX$SEV>/CFE -- Invalid DTE address/
150 000034 004 103 106 ERRLIN: .ASCIZ <EX$SEV>/CFE -- Error in line id/
151 000065 004 103 106 ERRHSH: .ASCIZ <EX$SEV>/CFE -- Invalid hash table size/
152 000125 004 103 106 ERRCT: .ASCIZ <EX$SEV>/CFE -- Invalid counter timer value/
153 000171 004 103 106 ERRSTA: .ASCIZ <EX$SEV>/CFE -- Invalid DTE state/
154 000223 004 103 106 ERRNET: .ASCIZ <EX$SEV>/CFE -- Error in network name/
155 .EVEN
156 .LIST BEX
157 STA: .BLKW 1 ; STATE VALUE

```



55  
 56  
 57  
 58  
 59  
 60  
 61  
 62  
 63  
 64  
 65  
 66  
 67  
 68  
 69  
 70  
 71  
 72  
 73

.SBTTL MACRO DEFINITIONS

\*\*\*\*\*  
 : LOCAL MACROS  
 \*\*\*\*\*

.MACRO PRINT TEXT  
 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT  
 CALL \$PRINT ; PRINT MESSAGE  
 .ENDM PRINT

\*\*\*\*\*  
 : MACRO CALLS  
 \*\*\*\*\*

.MCALL ISTAT\$,STAT\$,TRANS\$,FLTDF\$  
 FLTDF\$ ; DEFINE FILTER BLOCK OFFSETS

000000

|    |     |                                                 |
|----|-----|-------------------------------------------------|
| 4- | 46  | MACRO DEFINITIONS                               |
| 4- | 78  | Macro calls and local data                      |
| 5- | 128 | State tables                                    |
| 6- | 197 | \$TMSEB - Setup remote characteristics template |
| 7- | 233 | Action routines                                 |
| 8- | 351 | Utility subroutines                             |

TMPFIL      CREATED BY    MACRO    ON 29-JUN-85 AT 05:35      PAGE 2      I 7  
SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL | VALUE         | REFERENCES                             |
|--------|---------------|----------------------------------------|
| .FLAGS | =   *****  GX | *6-216                                 |
| .PSTCN | =   *****  GX | 7-259      7-278      8-364      8-366 |
| .PSTPT | =   *****  GX | 8-365      8-371                       |
| .TEMP  | =   *****  GX | *6-214      7-344                      |
| .TPARS | =   *****  GX | 6-222                                  |

|                  |                  |                     |                   |                   |
|------------------|------------------|---------------------|-------------------|-------------------|
| CS.FEA= ***** GX | FEAKTB 000000RG  | 003 \$ALPHA= 000022 | \$EXIT = 000000   | \$TMFEA 000002RG  |
| C.DEV 000004     | FEASTB 000000RG  | 002 \$ANY = 000020  | \$FAIL = 177777   | \$\$\$FLG= 177777 |
| C.LEN = 000014   | SETDV1 000056R   | \$BLANK= 000006     | \$LAMDA= 000000   | \$\$\$KEY= 177777 |
| C.STS = ***** GX | SETDV2 000072R   | \$CNFIG= ***** GX   | \$NUMBR= 000002   | \$\$\$STA= 000000 |
| C.WD1 000006     | SETWD1 000106R   | \$DIGIT= 000024     | \$RAD50= 000016   | .PCHAR= ***** GX  |
| C.WD2 000010     | SETWD2 000142R   | \$DNUMB= 000014     | \$STRNG= 000004   | .PNUMB= ***** GX  |
| C.WD3 000012     | SETWD3 000170R   | \$EOS = 000012      | \$SUBXP= 000010   | .PNUMH= ***** GX  |
| EXIT 000032R     | 002 TEMP 000000R | \$ERROR= ***** GX   | \$TALOC= ***** GX | .TPARS= ***** GX  |

. ABS. 000014 000 (RW,I,GBL,ABS,OVR)  
000216 001 (RW,I,LCL,REL,CON)  
\$STATE 000036 002 (RW,D,LCL,REL,CON)  
\$KTAB 000000 003 (RW,D,LCL,REL,CON)  
\$KSTR 000000 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10406 Words ( 41 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:17.14  
SY:TMPFEA.V2,[132,134]TMPFEA/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPFEA

SETLIN - SET LINE NUMBER

```

278 .SBTTL SETLIN - SET LINE NUMBER
279 ;+
280 ;*** - SETLIN - SETUP LINE NUMBER
281 ;
282 ; THIS ACTION ROUTINE CHECKS THE LLC LINE NUMBER TO MAKE SURE IT IS
283 ; VALID.
284 ;
285 ; INPUT:
286 ; .PNUMB = LINE NUMBER
287 ; .PNUMH = LINE NUMBER (HIGH ORDER)
288 ;
289 ; OUTPUT:
290 ; THE LINE NUMBER IS STORED IN C.LIN OF THE TEMPLATE.
291 ;
292 ;-
293
294 000222 SETLIN:
295 000222 016700 177554 MOV TEMP,R0 ; GET START OF TEMPLATE
296 000226 016767 000000G 000000G MOV .PNUMB,$NLN ; STORE THE NUMBER OF NSP CHANNELS
297 000234 005267 000000G INC $NLN
298 000240 116760 000000G 000011 MOVB .PNUMB,C.LIN(R0) ; STORE THE NUMBER OF LINES ON THIS LLC
299 000246 005767 000000G TST .PNUMH ; IS IT ONLY ONE WORD ?
300 000252 001003 BNE 10$; IF NE, NO - REJECT
301 000254 105767 000001G TSTB .PNUMB+1 ; IS IT A BYTE VALUE ?
302 000260 001402 BEQ 20$; IF EQ, YES - OKAY
303 000262 062716 000002 ADD #2,(SP) ; ELSE, REJECT TRANSITION
304 000266 000207 10$: RETURN
 20$:

```

SETDEV - STORE THE PSEUDO DEVICE NAME

```

72 .SBTTL LOCAL DATA
73 ;****
74 ; LOCAL DATA
75 ;****
76
77 000007 MAXSTA = 7 ; MAXIMUM VALUE FOR LOGGING STATE
78 000074 LA = '<' ; LEFT ANGLE BRACKET
79 000076 RA = '>' ; RIGHT ANGLE BRACKET
80 000000 TEMP: .BLKW 1 ; ADDRESS OF NODE TEMPLATE
81
82 ; DEFINE LOGGING STATE TEMPLATE
83 ;
84 000000 .ASECT
85 000000 .=0
86 000000 .BLKW 1 ; C.LNK
87 000002 .BLKW 1 ; C.STS
88 000004 C.STA: .BLKB 1 ; LOGGING STATE
89
90 000005 C.LEN:
91 000002 .PSECT

```

181  
182  
183  
184  
185  
186  
187  
188  
189  
190  
191  
192  
193  
194  
195  
196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213

000100  
000100 016700 000000G  
000104 010701  
000106 004767 000000G  
000112 016702 177662  
000116 010162 000004  
000122 010167 000000G  
000126 010701  
000130 004767 000000G  
000134 016702 177640  
000140 010162 000006  
000144 010167 000002G  
000150 022767 000006 000000G  
000156 103002  
000160 062716 000002  
000164 000207

```
.SBTTL SETNAM - SET UP NODE NAME
*** - SETNAM - SETUP NODE NAME
THIS ACTION ROUTINE SETS UP THE NODE NAME.
INPUT:
.PSTPT = START ADDRESS OF THE NODE NAME IN ASCII
.PSTCN = NUMBER OF CHARACTERS IN THE NODE NAME
TEMP = TEMPLATE ADDRESS
OUTPUT:
C.NAM = RAD50 NODE NAME (2 WORDS)
$NTNAM = NODE NAME
--
SETNAM:
MOV .PSTPT,R0 ; GET ADDRESS OF ASCII NODE NAME
MOV PC,R1 ; PERIODS ARE ACCEPTABLE
CALL $CAT5 ; CONVERT NODE NAME TO RAD50
MOV TEMP,R2 ; GET START OF TEMPLATE
MOV R1,C.NAM(R2) ; STORE NODE NAME IN TEMPLATE
MOV R1,$NTNAM ; ..TWICE
MOV PC,R1 ; PERIODS ARE ACCEPTABLE
CALL $CAT5 ; CONVERT THE REST OF THE NODE NAME
MOV TEMP,R2 ; GET START OF TEMPLATE
MOV R1,C.NAM+2(R2) ; STORE THE SECOND WORD
MOV R1,$NTNAM+2 ; ..TWICE
CMP #6,.PSTCN ; IS THE NODE NAME TOO BIG ?
BHS 10$; IF HIS, NO
ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
10$: RETURN
```

```

96 ;****
97 ; INITIALIZE TABLE GENERATION
98 ;****
99
100
101 000002 ISTAT$ OBJSTB,OBJKT8
102
103
104 000002 STATES$
105 000002 TRANS$ $NUMBR,,SETYPE
106
107 000002 STATES$
108 000002 TRANS$ <','>
109
110 000002 STATES$
111 000002 TRANS$ <','>,OBJ0
112 000002 TRANS$ LA
113
114 000002 STATES$
115 000002 TRANS$ $RAD50,,SETNAM
116
117 000002 STATES$
118 000002 TRANS$ RA
119
120 000002 STATES$
121 000002 TRANS$ <','>
122
123 000002 STATES$ OBJ0
124 000002 TRANS$ $NUMBR,,SETVfy
125
126 000002 STATES$
127 000002 TRANS$ <','>
128
129 000002 STATES$
130 000002 TRANS$ $NUMBR,,SETFLG
131
132 000002 STATES$
133 000002 TRANS$ $EOS,$EXIT
134 000002 TRANS$ <','>
135
136 000002 STATES$
137 000002 TRANS$ $NUMBR,,SETCPY
138
139 ;
140 ; EXIT STATE
141 ;
141 000002 STATES$ EXIT
142 000002 TRANS$ $EOS,$EXIT
143
144 000002 STATES$

```



.TITLE TMPPAR - CFE PARSE PARTITION DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1984, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE PARTITION DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

|            |     |     |          |          |          |      |      |      |
|------------|-----|-----|----------|----------|----------|------|------|------|
| TTTTTTTTTT | MM  | MM  | PPPPPPPP | PPPPPPPP | SSSSSSSS | NN   | NN   |      |
| TTTTTTTTTT | MM  | MM  | PPPPPPPP | PPPPPPPP | SSSSSSSS | NN   | NN   |      |
| TT         | MMM | MMM | PP       | PP       | SS       | NN   | NN   |      |
| TT         | MMM | MMM | PP       | PP       | SS       | NN   | NN   |      |
| TT         | MM  | MM  | PP       | PP       | SS       | NNNN | NN   |      |
| TT         | MM  | MM  | PP       | PP       | SS       | NNNN | NN   |      |
| TT         | MM  | MM  | PPPPPPPP | PPPPPPPP | SSSSSS   | NN   | NN   |      |
| TT         | MM  | MM  | PPPPPPPP | PPPPPPPP | SSSSSS   | NN   | NN   |      |
| TT         | MM  | MM  | PP       | PP       | SS       | NN   | NNNN |      |
| TT         | MM  | MM  | PP       | PP       | SS       | NN   | NNNN |      |
| TT         | MM  | MM  | PP       | PP       | SS       | NN   | NN   | .... |
| TT         | MM  | MM  | PP       | PP       | SS       | NN   | NN   | .... |
| TT         | MM  | MM  | PP       | PP       | SSSSSSSS | NN   | NN   | .... |
| TT         | MM  | MM  | PP       | PP       | SSSSSSSS | NN   | NN   | .... |

|            |          |            |
|------------|----------|------------|
| LL         | SSSSSSSS | TTTTTTTTTT |
| LL         | SSSSSSSS | TTTTTTTTTT |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SSSSSS   | TT         |
| LL         | SSSSSS   | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LL         | SS       | TT         |
| LLLLLLLLLL | SSSSSSSS | TT         |
| LLLLLLLLLL | SSSSSSSS | TT         |

TMPPSN      CREATED BY    MACRO    ON 29-JUN-85 AT 05:38      PAGE 3      1 15  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES                                                                                       |
|------------|--------------------------------------------------------------------------------------------------|
| CALLR      | #4-124                                                                                           |
| DBGTP\$    | #6-159    #6-174    #6-178    #6-184    #6-188    #6-194    #6-198    #6-204                     |
| EPRINT     | #4-78                                                                                            |
| ERROR\$    | #4-67    7-250    7-256                                                                          |
| ISTAT\$    | #4-124    6-159                                                                                  |
| MTRAN\$    | #6-159                                                                                           |
| PRINT      | #4-62                                                                                            |
| STAT\$     | #4-124    6-163    #6-165    #6-167    #6-170    #6-172    #6-176    #6-180    #6-182    #6-186  |
|            | #6-190    #6-192    #6-196    #6-200    #6-202    #6-206    #6-211                               |
| TRAN\$     | #4-124    #6-164    #6-166    #6-168    #6-171    #6-173    #6-174    #6-177    #6-178    #6-181 |
|            | #6-183    #6-184    #6-187    #6-188    #6-191    #6-193    #6-194    #6-197    #6-198    #6-201 |
|            | #6-203    #6-204    #6-207                                                                       |
| \$GTNMB    | #4-100                                                                                           |
| \$GTNUM    | #4-90    7-287    7-289    7-291    7-293                                                        |
| \$GTR50    | #4-112    7-279                                                                                  |

| SYMBOL  | VALUE      | REFERENCES |        |       |       |       |       |       |             |
|---------|------------|------------|--------|-------|-------|-------|-------|-------|-------------|
| CS.FLG  | = ***** GX | 7-332      |        |       |       |       |       |       |             |
| CS.PVC  | = ***** GX | 7-268      |        |       |       |       |       |       |             |
| C\$LENX | = ***** GX | 7-280      |        |       |       |       |       |       |             |
| DLM     | 000352 R   | #5-147     | *7-271 | 7-278 |       |       |       |       |             |
| ERRCT   | 000100 R   | #5-137     |        |       |       |       |       |       |             |
| ERRFLG  | 000273 R   | #5-141     |        |       |       |       |       |       |             |
| ERRLCN  | 000031 R   | #5-136     |        |       |       |       |       |       |             |
| ERRNAM  | 000000 R   | #5-135     | 7-284  |       |       |       |       |       |             |
| ERROWN  | 000235 "   | #5-140     |        |       |       |       |       |       |             |
| ERRPSZ  | 000137 H   | #5-138     |        |       |       |       |       |       |             |
| ERRSTA  | 000322 R   | #5-142     |        |       |       |       |       |       |             |
| ERRWSZ  | 000201 R   | #5-139     |        |       |       |       |       |       |             |
| EX\$SEV | = 000004   | #5-134     | 5-135  | 5-136 | 5-137 | 5-138 | 5-139 | 5-140 | 5-141 5-142 |
| PVCKTB  | 000000 RG  | #6-154     |        |       |       |       |       |       |             |
| PVCSTB  | 000000 RG  | #6-154     | 7-275  |       |       |       |       |       |             |
| PV\$OWN | = ***** GX | 7-330      |        |       |       |       |       |       |             |
| PV\$PSZ | = ***** GX | 7-326      |        |       |       |       |       |       |             |
| PV\$WSZ | = ***** GX | 7-328      |        |       |       |       |       |       |             |
| P\$VCT  | = ***** GX | 7-320      |        |       |       |       |       |       |             |
| P\$VFLG | = ***** GX | 7-332      |        |       |       |       |       |       |             |
| P\$VLCN | = ***** GX | 7-318      |        |       |       |       |       |       |             |
| P\$VLEN | = ***** GX | 7-263      |        |       |       |       |       |       |             |
| P\$VNAM | = ***** GX | 7-316      | 8-348  | 8-350 | 8-352 |       |       |       |             |
| P\$VOWN | = ***** GX | 7-282      | 7-330  |       |       |       |       |       |             |
| P\$VPSZ | = ***** GX | 7-326      |        |       |       |       |       |       |             |
| P\$VSTA | = ***** GX | *7-323     |        |       |       |       |       |       |             |
| P\$VWSZ | = ***** GX | 7-328      |        |       |       |       |       |       |             |
| SETCON  | 000764 R   | #8-350     |        |       |       |       |       |       |             |
| SETCT   | 000570 R   | #7-320     |        |       |       |       |       |       |             |
| SETDEV  | 000750 R   | #8-348     |        |       |       |       |       |       |             |
| SETFLG  | 000722 R   | #7-332     |        |       |       |       |       |       |             |
| SETLCN  | 000550 R   | #7-318     |        |       |       |       |       |       |             |
| SETNAM  | 000534 R   | #7-316     |        |       |       |       |       |       |             |
| SETOWN  | 000700 R   | #7-330     |        |       |       |       |       |       |             |
| SETPSZ  | 000624 R   | #7-326     |        |       |       |       |       |       |             |
| SETSTA  | 000610 R   | #7-322     |        |       |       |       |       |       |             |
| SETTRB  | 001004 R   | #8-352     |        |       |       |       |       |       |             |
| SETWSZ  | 000652 R   | #7-328     |        |       |       |       |       |       |             |
| STA     | 000350 R   | #5-146     | *7-270 | 7-323 |       |       |       |       |             |
| X\$LENX | = ***** GX | 7-272      |        |       |       |       |       |       |             |
| \$ALPHA | = 000022   | #6-154     |        |       |       |       |       |       |             |
| \$ANY   | = 000020   | #6-154     |        |       |       |       |       |       |             |
| \$BLANK | = 000006   | #6-154     |        |       |       |       |       |       |             |
| \$BYTMN | = ***** GX | *7-272     | *7-280 |       |       |       |       |       |             |
| \$CFERR | = ***** GX | 7-287      |        |       |       |       |       |       |             |
| \$DIGIT | = 000024   | #6-154     |        |       |       |       |       |       |             |
| \$DNUMB | = 000014   | #6-154     |        |       |       |       |       |       |             |
| \$EOS   | = 000012   | #6-154     |        |       |       |       |       |       |             |
| \$ERROR | = ***** GX | *7-288     | *7-289 |       |       |       |       |       |             |
| \$EXIT  | = 000000   | #6-154     |        |       |       |       |       |       |             |
| \$FAIL  | = 177777   | #6-154     |        |       |       |       |       |       |             |
| \$GPRM  | = *****    | 6-154      |        |       |       |       |       |       |             |

```

57 .SBTTL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61 .MACRO PRINT TEXT
62 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
63 CALL $PRINT ; PRINT MESSAGE
64 .ENDM PRINT
65
66 .MACRO ERRORS$ TEXT
67
68 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
69 ; MESSAGE STRING.
70
71 .IF DIF <TEXT><R0>
72 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
73 .ENDC
74 CALL $CFERR ; PRINT ERROR MESSAGE
75 .ENDM ERRORS$
76
77 .MACRO EPRINT TEXT
78
79 ; PRINT TEXT ON ERROR LUN
80
81 .IF DIF <TEXT><R0>
82 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
83 .ENDC
84 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
85 .ENDM EPRINT
86
87 ; GET NUMBER WITH RANGE CHECKING
88
89 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
90 MOV OFFS,R2
91 MOV LO,R0
92 MOV HI,R1
93 .IF NB <OPT>
94 BIS OPT,@.FLAGS
95 .ENDC
96 JMP $GTNUM
97 .ENDM
98
99 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
100 MOV OFFS,R2
101 MOV LO,R0
102 MOV HI,R1
103 .IF NB <OPT>
104 BIS OPT,@.FLAGS
105 .ENDC
106 JMP $GTNMB
107 .ENDM
108
109 ; GET RAD50 WITH FIELD SIZE CHECKING
110
111 .MACRO $GTR50 OFFS,LEN,OPT
112 MOV OFFS,R2
113

```

```
114 MOV LEN,R1
115 .IF NB <OPT>
116 BIS OPT,a.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 ;
122 ; MACRO CALLS
123 ;
124 .MCALL 1STAT$,STAT$,TRAN$,CALLR
```

```

LOCAL DATA
127
128
129
130
131 000000
132 000000
133 000000
134 000002
135 000004
136 000006
137 000012
138 000013
139 000014
140 000020
141 000000
142
143
144
145
146
147 000004
148 000000 004 103 106
149 000041 004 103 106
150 000102 004 103 106
151 000133 004 103 106
152 000171 004 103 106
153
154

.SBTTL LOCAL DATA
;
; DEFINE THE DST PARAMETER BLOCK
;
.ASECT
.=0
.BLKW 1 ; C.LNK
.BLKW 1 ; C.STS
C.TYP: .BLKW 1 ; RAD50 'X25' OR 'X29'
C.NAM: .BLKW 2 ; RAD50 DESTINATION NAME
C.PRI: .BLKB 1 ; LIST INSERTION PRIORITY
C.OBJ: .BLKB 1 ; OBJECT NUMBER
C.TSK: .BLKW 2 ; RAD50 TASK NAME
C.LEN=.
.PSECT
.NLIST BEX
;
; SYNTAX ERROR DETAIL STRINGS
;
.NLIST BEX
EX$SEV = 4
ERRTYP: .ASCIZ <EX$SEV>/CFE -- Illegal destination type/
ERRNAM: .ASCIZ <EX$SEV>/CFE -- Illegal destination name/
ERRPRI: .ASCIZ <EX$SEV>/CFE -- Illegal priority/
ERROBJ: .ASCIZ <EX$SEV>/CFE -- Illegal object number/
ERRTSK: .ASCIZ <EX$SEV>/CFE -- Illegal task name/
.EVEN
.LIST BEX

```

```

159 .SBTTL TPARS STATE TABLES
160 ;
161 ; INITIALIZE TABLES
162 ;
163
164 000264 ISTAT$ DTESTB, DTEKTB
165 ;
166 ; PROCESS 'DTE$DF'
167 ;
168
169 000264 STATES$; DTE ADDRESS
170 000264 TRANS$ $LAMDA,, $CLERR,ERRADR,,ERROR
171 000264 STATES$
172 000264 TRANS$!ADRSTR,,SETADR
173 000264 STATES$
174 000264 TRANS$ <','>,, $CLERR,ERRLIN,,ERROR
175
176
177 000264 STATES$; LINE ID
178 000264 TRANS$!LINJD
179 000264 STATES$
180 000264 TRANS$ <','>,, $CLERR,ERRHSH,,ERROR
181
182
183 000264 STATES$; HASH TABLE SIZE
184 000264 TRANS$ $NUMBR,,SETHSH
185 000264 STATES$
186 000264 TRANS$ <','>,, $CLERR,ERRCT,,ERROR
187
188
189 000264 STATES$; COUNTER TIMER VALUE
190 000264 TRANS$ $NUMBR,,SETCT
191 000264 STATES$
192 000264 TRANS$ <','>,, $CLERR,ERRSTA,,ERROR
193
194
195 000264 STATES$; STATE
196 000264 TRANS$ %ON%,DTE1,,1,STA
197 000264 TRANS$ %OFF%,DTE1
198 000264 STATES$ DTE1
199 000264 TRANS$ $EOS,$EXIT,SETSTA
200 000264 TRANS$ <','>,,SETSTA
201
202 000264 STATES$; NETWORK NAME
203 000264 TRANS$ $LAMDA,, $CLERR,ERRNET,,ERROR
204 000264 STATES$
205 000264 TRANS$ $STRNG,,SETNET
206 000264 STATES$
207 000264 TRANS$ $EOS,$EXIT
208
209 ;
210 ; SUBEXPRESSION TO PARSE DTE ADDRESS STRING
211 ;
212 000264 STATES$ ADRSTR
213 000264 TRANS$ $DIGIT
214 000264 STATES$ ADR1
215 000264 TRANS$ $DIGIT,ADR1

```



LOCAL DATA

```

75 .SBTTL LOCAL DATA
76 :****
77 : LOCAL DATA
78 :****
79
80 000074 LBRA = '<' ; LEFT ANGLE BRACKET
81 000076 RBRA = '>' ; RIGHT ANGLE BRACKET
82 077700 MAXCLS = 511.*100 ; MAXIMUM EVENT CLASS VALUE
83 000000 TEMP: .BLKW 1 ; ADDRESS OF NODE TEMPLATE
84 0000C2 STRADD: .BLKW 1 ; ADDRESS OF NEXT EVENT MASK TO STORE
85
86 :
87 : DEFINE EVENT TEMPLATE
88 :
89 000040 .ASECT
90 000000 .=0
91 000002 .BLKW 1 ; C.LNK
92 000004 .BLKW 1 ; C.STS
93 000006 C.CLS: .BLKW 1 ; EVENT CLASS
94 000016 C.EVT: .BLKW 4 ; EVENT MASKS
95 ; SINK TYPE
96 000020 C.SNK: .BLKW 1
97 000004 C.LEN=.
98 .PSECT

```

STATE TABLE

.TITLE TMPFIL - CFE PARSE REMOTE CHARACTERISTICS IN CETAB  
 .IDENT /V05.00/  
 .ENABL LC

Copyright (c) 1983, 1984, 1985 by  
 Digital Equipment Corporation, Maynard, Mass.

This software is furnished under a license and may be used and copied only in accordance with the terms of such license and with the inclusion of the above copyright notice. This software or any other copies thereof may not be provided or otherwise made available to any other person. No title to and ownership of the software is hereby transferred.

The information in this software is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation.

Digital assumes no responsibility for the use or reliability of its software on equipment which is not supplied by Digital.

Module description:

State table to parse the remote characteristics

Distributed Systems Software Engineering

Ident history:

4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0

5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0

TMPFIL      CREATED BY    MACRO    ON 29-JUN-85 AT 05:35      PAGE 3      J 7

MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGIP\$    | #5-132     | #5-138 | #5-139 | #5-145 | #5-146 | #5-152 | #5-153 | #5-159 | #5-160 | #5-166 |
|            | #5-173     | #5-179 | #5-183 | #5-184 | #5-188 | #5-192 | #5-193 |        |        |        |
| EPRINT     | #4-67      |        |        |        |        |        |        |        |        |        |
| ERROR\$    | #4-56      | 6-226  |        |        |        |        |        |        |        |        |
| ISTAT\$    | #4-80      | 5-132  |        |        |        |        |        |        |        |        |
| MTRAN\$    | #5-132     |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-51      |        |        |        |        |        |        |        |        |        |
| RESRG      | #4-80      |        |        |        |        |        |        |        |        |        |
| SAVRG      | #4-80      |        |        |        |        |        |        |        |        |        |
| STATE\$    | #4-80      | 5-134  | #5-136 | #5-141 | #5-143 | #5-148 | #5-150 | #5-155 | #5-157 | #5-162 |
|            | #5-164     | #5-171 | #5-177 | #5-181 | #5-186 | #5-190 | #5-195 |        |        |        |
| TRAN\$     | #4-80      | #5-135 | #5-137 | #5-138 | #5-139 | #5-142 | #5-144 | #5-145 | #5-146 | #5-149 |
|            | #5-151     | #5-152 | #5-153 | #5-156 | #5-158 | #5-159 | #5-160 | #5-163 | #5-165 | #5-166 |
|            | #5-172     | #5-173 | #5-178 | #5-179 | #5-182 | #5-183 | #5-184 | #5-187 | #5-188 | #5-191 |
|            | #5-192     | #5-193 |        |        |        |        |        |        |        |        |

TMPFEA CREATED BY MACRO ON 29-JUN-85 AT 05:35 PAGE 1 J 8  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL    | VALUE       | REFERENCES                                   |
|-----------|-------------|----------------------------------------------|
| CS.FEA    | = ***** GX  | 6-147                                        |
| C.DEV     | = 000004    | #4-82 *7-172 *8-190                          |
| C.LEN     | = 000014    | #4-86 6-144                                  |
| C.STS     | = ***** GX  | *6-147                                       |
| C.WD1     | = 000006    | #4-83 *9-211                                 |
| C.WD2     | = 000010    | #4-84 *10-234                                |
| C.WD3     | = 000012    | #4-85 *11-256                                |
| FEAKTB    | = 000000 RG | #5-95 6-149                                  |
| FEASTB    | = 000000 RG | #5-95 6-150                                  |
| SETDV1    | = 000056 R  | #7-170                                       |
| SETDV2    | = 000072 R  | #8-188                                       |
| SETWD1    | = 000106 R  | #9-209                                       |
| SETWD2    | = 000142 R  | #10-232                                      |
| SETWD3    | = 000170 R  | #11-254                                      |
| TEMP      | = 000000 R  | #4-74 *6-146 7-171 8-189 9-210 10-233 11-255 |
| \$ALPHA   | = 000022    | #5-95                                        |
| \$ANY     | = 000020    | #5-95                                        |
| \$BLANK   | = 000006    | #5-95                                        |
| \$CNFIG   | = ***** GX  | *9-212                                       |
| \$DIGIT   | = 000024    | #5-95                                        |
| \$DNUMB   | = 000014    | #5-95                                        |
| \$EOS     | = 000012    | #5-95                                        |
| \$ERROR   | = ***** GX  | *6-153                                       |
| \$EXIT    | = 000000    | #5-95                                        |
| \$FAIL    | = 177777    | #5-95                                        |
| \$GPRM    | = *****     | 5-95                                         |
| \$LAMDA   | = 000000    | #5-95                                        |
| \$NUMBR   | = 000002    | #5-95                                        |
| \$RAD50   | = 000016    | #5-95                                        |
| \$RONLY   | = *****     | 5-95 5-95                                    |
| \$STRNG   | = 000004    | #5-95                                        |
| \$SUBXP   | = 000010    | #5-95                                        |
| \$TALOC   | = ***** GX  | 6-145                                        |
| \$TMFEA   | = 000002 RG | #6-142                                       |
| \$\$\$FLG | = 177777    | #5-95                                        |
| \$\$\$KEY | = 177777    | #5-95                                        |
| .PCHAR    | = ***** GX  | 7-172 8-190                                  |
| .PNUMB    | = ***** GX  | 9-211 9-212 10-234 11-256                    |
| .PNUMH    | = ***** GX  | 9-213 10-235 11-257                          |
| .TPARS    | = ***** GX  | 6-151                                        |

```

306 .SBTTL SETDEV - STORE THE PSEUDO DEVICE NAME
307 ;+
308 *** - SETDEV - STORE THE PSEUDO DEVICE NAME
309 ;
310 INPUT:
311 .PSTPT = START OF THE ASCII DEVICE NAME
312 .PSTCN = COUNT OF THE CHARS IN THE DEVICE NAME
313 ;
314 OUTPUT:
315 C-BIT = SUCCESS/FAILURE
316 THE DEVICE NAME IS STORED IN THE TMLATE AT C.DEV
317 ;
318 ;
319 ;
320 000270 SETDEV:
321 000270 016700 177506 MOV TEMP,R0 ; GET START OF TEMPLATE
322 000274 116760 000000G 000012 MOVB .PSTPT,C.DEV(R0) ; STORE THE DEVICE NAME
323 000302 116760 000001G 000013 MOVB .PSTPT+1,C.DEV+1(R0) ;
324 000310 022767 000002 000000G CMP #2,.PSTCN ; IS THE DEVICE NAME TOO BIG ?
325 000316 103002 BHS 10$; IF HIS, NO - OKAY
326 000320 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION (SETS C-BIT)
327 000324 000207 10$: RETURN

```

```

STATE TABLE
93
94 .SBTTL STATE TABLE
95 ;****
96 ; INITIALIZE TABLE GENERATION
97 ;****
98
99 000002 ISTAT$ LOGSTB,LOGKTB
100
101
102 000002 STATES
103 000002 TRANS $NUMBR,,SETSTA
104
105 ;
106 ; EXIT STATE
107 ;
108 000002 STATES EXIT
109 000002 TRANS $EOS,$EXIT
110
111 000002 STATES

```

```

215 .SBTTL SETNID - SET UP NODE ID
216
217 :+
218 :*** - SETNID - SETUP NODE ID
219 :
220 :THIS ACTION ROUTINE SETS UP THE NODE ID.
221 :
222 :INPUT:
223 :.PSTPT = START ADDRESS OF THE NODE ID
224 :.PSTCN = NUMBER OF CHARACTERS IN THE NODE ID
225 :TEMP = TEMPLATE ADDRESS
226 :
227 :OUTPUT:
228 :C.NID = LENGTH OF NODE ID FOLLOWED BY THE NODE ID.
229 :-
230
231 SETNID:
232 000166 MOV TEMP,R0 ; GET START ADDRESS OF WHERE TO STORE ID
233 000166 016700 177606 ADD #C.NID,R0 ;
234 000172 062700 000010 MOV .PSTPT,R1 ; GET START OF INPUT STRING
235 000176 016701 000000G MOV .PSTCN,R2 ; AND LENGTH OF STRING
236 000202 016702 000000G DEC R2 ; SKIP THE '>'
237 000210 010220 MOV R2,(R0)+ ; STORE THE NODE ID LENGTH
238 000212 001402 BEQ 10$; IF EQ, ZERO LENGTH STRING
239 000214 004767 000000G CALL $MVASC ; MOVE THE NODE ID INTO THE TEMPLATE
240 000220 022767 000041 000000G 10$: CMP #33,..PSTCN ; IS THE NODE ID TOO BIG ?
241 000226 103002 BHS 20$; IF HIS, NO
242 000230 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
243 000234 000207 20$: RETURN

```

```

146 .SBTTL $TMOBJ - SETUP OBJECT TEMPLATE
147
148 *** - $TMOBJ - SETUP OBJECT TEMPLATE
149
150 INPUT:
151 R3-R5 - TPARS REGISTERS
152
153 OUTPUT:
154 C-BIT = SUCCESS/FAILURE
155 IF SUCCESS, THE OBJECT TEMPLATE IS STORED IN THE END OF TASK BUFFER.
156
157
158
159 $TMOBJ::
160 000002 MOV R5,-(SP) ; SAVE R5
161 000004 010546 MOV #C.LEN,R1 ; GET LENGTH OF ALLOCATION
162 000010 004767 000013 CALL $TALC ; TRY TO ALLOCATE CORE BLOCK
163 000014 010067 177760 MOV R0,TEMP ; STORE TEMPLATE ADDRESS
164 000020 012760 000000G 000000G MOV #CS.OBJ,C.STS(R0) ; INDICATE THAT THIS IS AN OBJ$DF TEMPLATE
165 000026 005060 000006 CLR C.NAM(R0) ; ASSUME OBJECT TYPE ZERO - NO NAME
166 000032 005060 000010 CLR C.NAM+2(R0) ;
167 000036 112760 000001 000012 MOVB #1,C.CPY(R0) ; ASSUME COPIES FIELD IS ABSENT
168 000044 005001 CLR R1 ; IGNORE BLANKS
169 000046 012702 000000* MOV #OBJKTBL,R2 ; KEYWORD TABLE
170 000052 012705 000000* MOV #OBJSTBL,R5 ; STATE TABLE
171 000056 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
172 000062 103002 BCC 20$; NORMAL RETURN IF NO ERROR
173 000064 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
174 000070 016700 177704 20$: MOV TEMP,R0 ; RETRIEVE TEMPLATE ADDRESS
175 000074 004767 000016 CALL FINOBJ ; FIND PLACE FOR THE OBJECT DEFINITION
176 000100 004767 000000G CALL $QINSN ; INSERT ENTRY INTO QUEUE
177 000104 062767 000000G 000000G ADD #0,LENX,$BYTMN ; UPDATE MINIMUM POOL BYTE EXTENSION
178 000112 012605 MOV (SP)+,R5 ; RESTORE R5
179 000114 000207 RETURN

```



```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 ;****
65 ; MACRO CALLS
66 ;****
67
68 .MCALL ISTAT$,STAT$,TRANS$
69
70 ;****
71 ; LOCAL DATA
72 ;****
73
74 000000 FLAG: .BLKW 1 ; FLAGS WORD
75 000002 TEMP: .BLKW 1 ; ADDRESS OF DLC TEMPLATE
76
77 ; DEFINE PARTITION TEMPLATE
78 ;
79 000000 .ASECT
80 000000 .=0
81 000000 .BLKW 1 ; C.LNK
82 000002 .BLKW 1 ; C.STS
83 000004 C.NAM0: .BLKW 1 ; POOL PARTITION NAME (RAD50)
84 000006 .BLKW 1
85 000010 C.NAM1: .BLKW 1 ; DYNAMIC PARTITION TO CREATE POOL FROM
86 000012 .BLKW 1 ; (2 WORDS - RAD50)
87 000014 C.EXT: .BLKW 1 ; POOL EXTENSION SIZE
88 000016 C.ALL: .BLKW 1 ; NUMBER OF BLOCKS TO ALLOCATE FOR SINGLE WORD STORAGE
89 000020 C.LEN=. .PSECT
90 000004
91

```

|    |     |                                                 |
|----|-----|-------------------------------------------------|
| 4- | 57  | MACRO DEFINITIONS                               |
| 5- | 126 | LOCAL DATA                                      |
| 6- | 154 | TPARS STATE TABLES                              |
| 7- | 214 | \$TMPPSN - BUILD PSN TEMPLATE                   |
| 7- | 261 | SETNAM - SET PSN NAME IN TEMPLATE               |
| 7- | 262 | SETPOR - SET NUMBER OF PORTS IN TEMPLATE        |
| 7- | 263 | SETFLG - SET FLAGS IN TEMPLATE                  |
| 7- | 264 | SETTHI - SET TRANSPORT HIGH ADDRESS IN TEMPLATE |
| 7- | 265 | SETTLO - SET TRANSPORT LOW ADDRESS IN TEMPLATE  |



IMPPVC      CREATED BY    MACRO    ON 29-JUN-85 AT 05:39      PAGE 2      J 16

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE       | REFERENCES                                         |
|-----------|-------------|----------------------------------------------------|
| \$GTNMB   | = ***** GX  | 7-332      8-350      8-352                        |
| \$GTNUM   | = ***** GX  | 7-318      7-320      7-326      7-328             |
| \$GTR50   | = ***** GX  | 7-316      7-330      8-348                        |
| \$LAMDA   | = 000000    | #6-154                                             |
| \$NUMBR   | = 000002    | #6-154                                             |
| \$RAD50   | = 000016    | #6-154                                             |
| \$RONLY   | = *****     | 6-154      6-154                                   |
| \$STRNG   | = 000004    | #6-154                                             |
| \$SUBXP   | = 000010    | #6-154                                             |
| \$TALOC   | = ***** GX  | 7-264                                              |
| \$TMPVC   | = 000354 RG | #7-261                                             |
| \$\$\$FLG | = 177777    | #6-154                                             |
| \$\$\$KEY | = 177777    | #6-154                                             |
| .ERROR    | = ***** GX  | *7-269      7-284      7-285                       |
| .FLAGS    | = ***** GX  | *7-267      7-326      7-328      7-330      7-332 |
| .TEMP     | = ***** GX  | *7-265      7-281      7-322                       |
| .TPARS    | = ***** GX  | 7-276                                              |

```

114 MOV LEN,R1
115 .IF NB <OPT>
116 BIS OPT,a.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 ;
122 ; MACRO CALLS
123 ;
124 .MCALL ISTAT$,STAT$,TRAN$,CALLR

```

```

LOCAL DATA
126
127
128
129
130
131
132 000000 000004 103 106
133 000032 004 103 106
134 000065 004 103 106
135
136

 .SBTTL LOCAL DATA
 .NLIST BEX

 :: SYNTAX ERROR DETAIL STRINGS
 ::
 EX$SEV = 4
 ERRMSK: .ASCIZ <EX$SEV>/CFE -- Illegal call mask/
 ERRVAL: .ASCIZ <EX$SEV>/CFE -- Illegal call value/
 ERRLEN: .ASCIZ <EX$SEV>/CFE -- Call mask and call value are different lengths/
 .EVEN
 .LIST BEX

```

```

156 .SBTTL TPARS STATE TABLES
157 ;
158 ; INITIALIZE TABLES
159 ;
160 ;
161 000224 ISTAT$ DSTSTB, DSTKTB
162 ;
163 ; PROCESS 'DST$DF'
164 ;
165 000224 STATES$; TYPE
166 000224 TRANS$ $LAMDA,, $CLERR,ERRTYP,.ERROR
167 000224 STATES$
168 000224 TRANS$ %X25%,DST1,SETTYP
169 000224 TRANS$ %X29%,DST1,SETTYP
170 000224 STATES$ DST1
171 000224 TRANS$ <','>,, $CLERR,ERRNAM,.ERROR
172
173
174 000224 STATES$; NAME
175 000224 TRANS$ $STRNG,,SETNAM
176 000224 STATES$
177 000224 TRANS$ <','>,, $CLERR,ERRPRI,.ERROR
178
179
180 000224 STATES$; PRIORITY
181 000224 TRANS$ $NUMBR,,SETPRI
182 000224 STATES$
183 000224 TRANS$ <','>,, $CLERR,ERROBJ,.ERROR
184
185
186 000224 STATES$; OBJECT
187 000224 TRANS$ $NUMBR,,SETOBJ
188 000224 STATES$
189 000224 TRANS$ $EOS,$EXIT
190 000224 TRANS$ <','>,, $CLERR,ERRTSK,.ERROR
191
192
193 000224 STATES$; OPTIONAL TASKNAME
194 000224 TRANS$ $RAD50,,SETTSK
195 000224 STATES$
196 000224 TRANS$ $EOS,$EXIT
197
198
199 ; FINAL STATE
200 ;
201 000224 STATES$
202

```

```

216 000264 TRANS $LAMDA,$EXIT
217 :
218 : SUBEXPRESSION TO PARSE LINE ID
219 :
220 000264 STATES LINID ;DEVICE NAME
221 000264 TRANS $STRNG,,SETDEV
222 000264 STATES LIN1
223 000264 TRANS <'->
224 000264 STATES ;CONTROLLER NUMBER
225 000264 TRANS $DNUMB,,SETCON
226 000264 STATES
227 000264 TRANS <'>.,LIN3
228 000264 TRANS <'>.,LIN4
229 000264 TRANS $LAMDA,$EXIT
230 000264 STATES LIN3
231 000264 TRANS $DNUMB,,SETUNI ;OPTIONAL UNIT NUMBER
232 000264 STATES
233 000264 TRANS <'>.,LIN4
234 000264 TRANS $LAMDA,$EXIT
235 000264 STATES LIN4 ;OPTIONAL TRIBUTARY NUMBER
236 000264 TRANS $DNUMB,$EXIT,SETTRB
237 :
238 : FINAL STATE
239 :
240 000264 STATES$
241

```



```

STATE TABLE

100
101
102
103
104
105
106
107 000004
108
109
110 000004
111 000004
112
113 000004
114 000004
115
116 000004
117 000004
118
119 000004
120 000004
121
122 000004
123 000004
124
125 000004
126 000004
127 000004
128 000004
129
130 000004
131 000004
132
133
134
135
136 000004
137 000004
138
139 000004
140 000004
141 000004
142
143 000004

.SBTTL STATE TABLE

;****
; INITIALIZE TABLE GENERATION
;****

ISTAT$ EVTSTB,EVTKTB

STATES$
TRANS$ $NUMBR,,SETCLS

STATES$
TRANS$ <','>

STATES$
TRANS$ LBRA

STATES$
TRANS$!EVTMSK

STATES$
TRANS$ <','>

STATES$
TRANS$ "CONSOLE",EXIT,SETCON
TRANS$ "FILE",EXIT,SETFIL
TRANS$ "MONITOR",,SETMON

STATES$ EXIT
TRANS$ $EOS,$EXIT

;
; EVENT MASK SUB-EXPRESSION
;
STATES$ EVTMSK
TRANS$ $NUMBR,,SETMSK

STATES$
TRANS$ RBRA,$EXIT
TRANS$ <','>,EVTMSK

STATES$

```

```

46 .SBTTL MACRO DEFINITIONS
47
48 ; LOCAL MACROS
49
50
51 .MACRO PRINT TEXT
52 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
53 CALL $PRINT ; PRINT MESSAGE
54 .ENDM PRINT
55
56 .MACRO ERROR$ TEXT
57
58 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
59 ; MESSAGE STRING.
60
61 .IF DIF <TEXT><RO>
62 MOV TEXT,RO ; GET ADDRESS OF ERROR MESSAGE
63 .ENDC
64 CALL $CFERR ; PRINT ERROR MESSAGE
65 .ENDM ERROR$
66
67 .MACRO EPRINT TEXT
68
69 ; PRINT TEXT ON ERROR LUN
70
71 .IF DIF <TEXT><RO>
72 MOV TEXT,RO ; GET ADDRESS OF ASCIZ STRING TO PRINT
73 .ENDC
74 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
75 .ENDM EPRINT
76
77 ;
78
79 .SBTTL Macro calls and local data
80
81 .MCALL ISTAT$,STATE$,TRAN$,SAVRG,RESRG
82
83 ; Define FIL$DF template
84
85 .ASECT
86 . = 0
87 .BLKW 1 ; Link
88 .BLKW 1 ; Flags
89 F$SEC: .BLKW 1 ; Secondary loader file name
90 F$TER: .BLKW 1 ; Tertiary loader file name
91 F$LD: .BLKW 1 ; Load file name
92 F$DIA: .BLKW 1 ; Diagnostics file name
93 F$DUM: .BLKW 1 ; Dump file name
94 F$LEN=.
95 .PSECT
96
97 ; Define file descriptor offsets
98
99
100 MXFSP = 30. ; Filespec string length
101
102

```

\*\*FILE\*\*ID\*\*TMPFEA

```

TTTTTTTTT MM MM PPPPPPP FFFFFFFF EEEEEEEEE AAAAAA
TTTTTTTTT MM MM PPPPPPP FFFFFFFF EEEEEEEEE AAAAAA
TT MMMM MMMM PP PP FF AA AA
TT MMMM MMMM PP PP FF AA AA
TT MM MM MM PP PP FF AA AA
TT MM MM MM PPPPPPP FFFFFFFF EEEEEEEEE AA AA
TT MM MM PPPPPPP FFFFFFFF EEEEEEEEE AA AA
TT MM MM PP FF AA AAAAAAAAAA
TT MM MM PP FF AA AAAAAAAAAA
TT MM MM PP FF AA AA
TT MM MM PP FF AA AA
TT MM MM PP FF AA AA
TT MM MM PP FF AA AA

```

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSS TT
LL SSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLL SSSSSSS TT
LLLLLLLLL SSSSSSS TT

```

TMPFEA      CREATED BY    MACRO    ON 29-JUN-85 AT 05:35      PAGE 2      K 8

MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |       |        |        |        |        |        |        |        |        |
|---------|--------|-------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$ | #5-95  |       |        |        |        |        |        |        |        |        |
| ISTAT\$ | #4-68  | 5-95  |        |        |        |        |        |        |        |        |
| MTRAN\$ | #5-95  |       |        |        |        |        |        |        |        |        |
| PRINT   | #4-59  |       |        |        |        |        |        |        |        |        |
| STATE\$ | #4-68  | 5-98  | #5-101 | #5-104 | #5-107 | #5-110 | #5-113 | #5-116 | #5-119 | #5-124 |
|         | #5-127 |       |        |        |        |        |        |        |        |        |
| TRAN\$  | #4-68  | #5-99 | #5-102 | #5-105 | #5-108 | #5-111 | #5-114 | #5-117 | #5-120 | #5-125 |

```

329 .SBTTL SETNLT - SET NUMBER OF LINE TABLES TO ALLOCATE SPACE FOR
330 :+
331 :*** - SETNLT - NUMBER OF LINE TABLE TO ALLOCATE SPACE
332 :
333 :INPUTS:
334 :.PNUMB = NUMBER OF LINE TABLES TO ALLOCATE SPACE
335 :TEMP = TEMPLATE ADDRESS
336 :
337 :OUTPUTS:
338 :NUMBER IS STORE IN C.NLT OF THE TEMPLATE
339 :TRANSITION IS ACCEPTED/REJECTED
340 :
341 :-
342
343 SETNLT:
344 000326 016700 177450 MOV TEMP,R0 ; GET THE TEMPLATE ADDRESS
345 000326 116760 000000G 000014 MOVB .PNUMB,C.NLT(R0) ; STORE THE NUMBER OF LINE TABLES
346 000340 026727 000000G 000000G CMP .PNUMB,#$MXNLT ; IS THE NUMBER IN RANGE ?
347 000346 101402 BLOS 10$; IF LOS, YES .. OKAY
348 000350 062716 ADD #2,(SP) ; ELSE, REJECT
349 000354 000207 10$: RETURN

```

TMPLOG - CFE PARSE LOGGING STAT MACRO V05.03b Saturday 29-Jun-85 05:36 Page 7  
 \$TMLOG - SETUP LOGGING TEMPLATE

```

113 .SBTTL $TMLOG - SETUP LOGGING TEMPLATE
114 :+
115 *** - $TMLOG - SETUP LOGGING TEMPLATE
116 :
117 INPUT:
118 R3-R5 - TPARS REGISTERS
119 :
120 OUTPUT:
121 C-BIT = SUCCESS/FAILURE
122 IF SUCCESS, THE OBJECT TEMPLATE IS STORED IN THE END OF TASK BUFFER.
123 :
124 :-
125
126 000002 $TMLOG::
127 000002 010546 MOV R5,-(SP) ; SAVE R5
128 000004 012701 000005 MOV #C.LEN,R1 ; GET LENGTH OF ALLOCATION
129 000010 004767 000000G CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
130 000014 010067 177760 MOV R0,TEMP ; STORE TEMPLATE ADDRESS
131 000020 012760 000000G 000000G MOV #CS.LOG,C.STS(R0) ; INDICATE THAT THIS IS A LOG$ST TEMPLATE
132 000026 012702 000000' MOV #LOGKTBL,R2 ; KEYWORD TABLE
133 000032 012705 000000' MOV #LOGSTBL,R5 ; STATE TABLE
134 000036 005001 CLR R1 ; IGNORE BLANKS
135 000040 004767 000000G CALL .TPARS ; PARSE THE REST OF THE LINE
136 000044 103002 BCC 20$; NORMAL RETURN IF NO ERROR
137 000046 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
138 000052 012605 20$: MOV (SP)+,R5 ; RESTORE R5
139 000054 000207 RETURN

```

TMPLOG - CFE PARSE LOGGING STAT MACRO V05.03b Saturday 29-Jun-85 05:36 Page 8  
 SETSTA - SET UP LOGGING STATE

```

245
246
247
248
249
250
251
252
253
254
255
256
257 000236
258 000236 016700 177536
259 000242 016701 000000G
260 000246 001416
261 000250 020127 000077
262 000254 101013
263 000256 000301
264 000260 006301
265 000262 006301
266 000264 010160 000052
267 000270 010167 000000G
268 000274 052760 000000G 000000G
269 000302 000402
270 000304 062716 000002
271 000310 000207
272

 +
 *** - SETNNA - SET NODE NUMBER AREA
 :
 : INPUT:
 : .PNUMB = NODE NUMBER
 : .PNUMH = NODE NUMBER (HIGH ORDER)
 :
 : OUTPUT:
 : NUMBER IS STORED IN C.NUM OF TEMPLATE
 :
 : -
 :
 SETNNA:
 MOV TEMP,R0 ; POINT R0 AT TEMPLATE
 MOV .PNUMB,R1 ; PICK UP AREA NUMBER
 BEQ 5$; BR IF TOO LOW
 CMP R1,#63. ; TOO HIGH ?
 BHI 5$; IF HI, ERROR
 SWAB R1 ; DIDDLE WITH FORMAT
 ASL R1 ; ...
 ASL R1 ;
 MOV R1,C.NUM(R0) ; STORE THE NODE NUMBER
 MOV R1,$NTADD ; ... TWICE
 BIS #NO$EXA,C.STS(R0) ; REMEMBER WE SAW AN AREA
 BR 10$
 5$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
 10$: RETURN

```

K 12

```

181 .SBTTL FINOBJ - FIND NEXT OBJECT IN SORTED LIST
182
183 :+
184 :*** - FINOBJ - FIND NEXT OBJECT IN SORTED LIST
185
186 :INPUTS:
187 :R0 = TEMPLATE TO BE INSERTED
188
189 :OUTPUTS:
190 :R1 = ADDRESS OF TEMPLATE TO PRECEED CURRENT TEMPLATE
191 :
192
193 000116 FINOBJ: MOV #$TMLST,R2 ; GET START ADDRESS OF TEMPLATES
194 000116 012702 000000G 10$: MOV R2,R1 ; SAVE ADDRESS OF PREVIOUS
195 000122 010201 : (R2),R2 ; GET NEXT TEMPLATE IN LIST
196 000124 011202 : BEQ 20$; IF EQ, END OF LIST
197 000126 001410 : CMPB #CS.OBJ,C.STS(R2) ; IS THIS A OBJECT TEMPLATE ?
198 000130 122762 000000G 000000G : BNE 10$; IF NE, NO .. KEEP LOOKING
199 000136 001371 : CMPB C.TYP(R0),C.TYP(R2) ; ELSE, DOES TEMPLATE GO HERE ?
200 000140 126062 000004 000004 : BHIS 10$; IF HIS, NO .. KEEP LOOKING
201 000146 103365 :
202 000150 000207 20$: RETURN ; ELSE, OKAY

```

L 12



K 13

```

93 ;****
94 ; INITIALIZE TABLE GENERATION
95 ;****
96
97 000004 ISTAT$ PARSTB,PARKTB
98
99 000004 STATES$
100 000004 TRANS$ $RAD50,NEXT,SETNMO
101 000004 TRANS$ $LAMDA
102
103 000004 STATES$ NEXT
104 000004 TRANS$ <','>
105
106 000004 STATES$
107 000004 TRANS$ $RAD50,,SETNM1
108
109 000004 STATES$
110 000004 TRANS$ <','>
111
112 000004 STATES$
113 000004 TRANS$ $NUMBR,,SETEXT
114
115 000004 STATES$
116 000004 TRANS$ <','>,TOP
117 000004 TRANS$ $LAMDA,EXIT
118
119 000004 STATES$ TOP
120 000004 TRANS$ "TOP",UMRS,,CS.TOP,FLAG
121 000004 TRANS$ $LAMDA
122
123 000004 STATES$ UMRS
124 000004 TRANS$ $EOS,EXIT
125 000004 TRANS$ <','>
126
127 000004 STATES$
128 000004 TRANS$ $NUMBR,ALLC
129 000004 TRANS$ "NONE",ALLC
130 000004 TRANS$ $LAMDA
131
132 000004 STATES$ ALLC
133 000004 TRANS$ <','>,ALL
134 000004 TRANS$ $LAMDA,EXIT
135
136 000004 STATES$ ALL
137 000004 TRANS$ $NUMBR,SBCHK,SETALL
138
139 000004 STATES$ SBCHK
140 000004 TRANS$ $EOS,EXIT
141 000004 TRANS$ <','>
142
143 000004 STATES$
144 000004 TRANS$ "SBPOOL",EXIT
145
146 ;
147 ; EXIT STATE
148 ;
149 000004 STATES$ EXIT

```

L 13

.TITLE TMPPSN - PARSE PSN IN CETAB  
.IDENT /V05.00/  
.ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE PSN DEFINITION

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

1.00 14-AUG-81  
DECNET-11M/S V3.1  
DECNET-11M-PLUS V1.1  
  
3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1  
  
4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0  
  
5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

|    |     |                                           |
|----|-----|-------------------------------------------|
| 4- | 57  | MACRO DEFINITIONS                         |
| 5- | 128 | LOCAL DATA                                |
| 6- | 149 | TPARS STATE TABLES                        |
| 7- | 246 | \$TMPVC - BUILD PVC TEMPLATE              |
| 7- | 293 | SETNAM - SET PVC NAME                     |
| 7- | 294 | SETLCN - SET LOGICAL CHANNEL NUMBER       |
| 7- | 295 | SETCT - SET COUNTER TIMER                 |
| 7- | 296 | SETPSZ - SET MAXIMUM BLOCK SIZE           |
| 7- | 297 | SETWSZ - SET WINDOW SIZE                  |
| 7- | 298 | SETOWN - SET OWNER PROCESS                |
| 7- | 299 | SETFLG - SET FLAGS BYTE                   |
| 7- | 300 | SETSTA - SET PVC STATE                    |
| 8- | 335 | SETDEV - SET DEVICE NAME IN LINE ID       |
| 8- | 336 | SETCON - SET CONTROLLER NUMBER IN LINE ID |
| 8- | 337 | SETTRB - SET TRIB NUMBER IN LINE ID       |

TMPPVC CREATED BY MACRO ON 29-JUN-85 AT 05:39 PAGE 3 K 16  
 MACRO CROSS REFERENCE CREF 04.00

| MACRO NAME | REFERENCES                                                            |
|------------|-----------------------------------------------------------------------|
| CALLR      | #4-124                                                                |
| DBGTP\$    | #6-154 #6-162 #6-184 #6-187 #6-193 #6-197 #6-203 #6-207 #6-213 #6-217 |
| EPRINT     | #4-78                                                                 |
| ERROR\$    | #4-67 7-287                                                           |
| ISTAT\$    | #4-124 6-154                                                          |
| MTRAN\$    | #6-154                                                                |
| PRINT      | #4-62                                                                 |
| STATF\$    | #4-124 6-158 #6-160 #6-163 #6-166 #6-168 #6-170 #6-173 #6-175 #6-177  |
|            | #6-180 #6-182 #6-185 #6-189 #6-191 #6-195 #6-199 #6-201 #6-205 #6-209 |
|            | #6-211 #6-215 #6-219 #6-221 #6-223 #6-229 #6-231 #6-233 #6-235 #6-237 |
|            | #6-243                                                                |
| TRANS      | #4-124 #6-159 #6-161 #6-162 #6-164 #6-167 #6-169 #6-171 #6-174 #6-176 |
|            | #6-178 #6-181 #6-183 #6-184 #6-186 #6-187 #6-190 #6-192 #6-193 #6-196 |
|            | #6-197 #6-200 #6-202 #6-203 #6-206 #6-207 #6-210 #6-212 #6-213 #6-216 |
|            | #6-217 #6-220 #6-222 #6-224 #6-230 #6-232 #6-234 #6-236 #6-238        |
| \$GTNMB    | #4-100 7-332 8-350 8-352                                              |
| \$GTNUM    | #4-90 7-318 7-320 7-326 7-328                                         |
| \$GTR50    | #4-112 7-316 7-330 8-348                                              |

```

LOCAL DATA
126 .SBTIL LOCAL DATA
127 000000 .PSECT
128 .NLIST BEX
129
130 ; SYNTAX ERROR DETAIL STRINGS
131 ;
132 EX$SEV = 4
133 000000 000004 103 106 ERRNUM: .ASCIZ <EX$SEV>/CFE -- Illegal destination address/
134 000044 004 103 106 ERRSLO: .ASCIZ <EX$SEV^>/CFE -- Illegal subaddress low range/
135 000111 004 103 106 ERRSHI: .ASCIZ <EX$SEV>/CFE -- Illegal subaddress high range/
136 000157 C04 103 106 ERRCUG: .ASCIZ <EX$SEV>/CFE -- Illegal CUG name/
137 .EVEN
138 .LIST BEX

```

```

138 .SBTTL TPARS STATE TABLES
139 ;
140 ; INITIALIZE TABLES
141 ;
142 ;
143 000154 ISTAT$ DSCSTB, DSCSTB
144 ;
145 ; PROCESS 'DSC$DF'
146 ;
147 ;
148 000154 STATES ; CALL MASK
149 000154 TRANS $LAMDA,, $CLERR, ERRMSK, .ERROR
150 000154 STATES
151 000154 TRANS $EOS, $EXIT
152 000154 TRANS !CALL,, SETMSK
153 000154 STATES
154 000154 TRANS <','>
155 ;
156 000154 STATES DSC1 ; CALL VALUE
157 000154 TRANS $LAMDA,, $CLERR, ERRVAL, .ERROR
158 000154 STATES
159 000154 TRANS $EOS, $EXIT
160 000154 TRANS !CALL,, SETVAL
161 000154 STATES
162 000154 TRANS $EOS, $EXIT
163 ;
164 ; SUBEXPRESSION TO PARSE CALL STRING
165 ;
166 000154 STATES CALL
167 000154 TRANS !HEXDIG
168 000154 STATES CALL1
169 000154 TRANS !HEXDIG,CALL1
170 000154 TRANS $LAMDA, $EXIT
171 ;
172 000154 STATES HEXDIG
173 000154 TRANS $DIGIT, $EXIT
174 000154 TRANS <'A>, $EXIT
175 000154 TRANS <'B>, $EXIT
176 000154 TRANS <'C>, $EXIT
177 000154 TRANS <'D>, $EXIT
178 000154 TRANS <'E>, $EXIT
179 000154 TRANS <'F>, $EXIT
180 ;
181 ; FINAL STATE
182 ;
183 000154 STATES
184

```

```

204 .SBTTL $TMDST - BUILD DST TEMPLATE
205
206 +
207 *** - $TMDST - BUILD DST TEMPLATE
208
209 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
210
211 INPUT:
212 R3-R5 - TPARS REGISTERS
213
214 OUTPUT:
215 C-BIT = SUCCESS/FAILURE
216 THE DST TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
217
218 -
219
220 $TMDST::
221 000224 MOV R5,-(SP) ; SAVE R5
222 000224 010546 MOV #C.LEN,R1 ; GET LENGTH OF NEEDED ALLOCATION
223 000226 012701 000020 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
224 000232 004767 000000G MOV RO,TEMP ; SAVE TEMPLATE ADDRESS
225 000236 010067 000000G TST (R0)+ ; POINT TO C.STS
226 000242 005720 MOV RO,.FLAGS ; SAVE FOR OPTION SETTING
227 000244 010067 000000G MOV #CS.DST,(R0) ; INDICATE THAT THIS IS A DST$DF TEMPLATE
228 000250 012710 000000G ADD #D$FLNX,$BYTMN ; UPDATE MINIMUM ALLOCATION
229 000254 062767 000000G CLR .ERROR ; NO ERROR DETAIL YET
230 000262 005067 000000G CLR R1 ; IGNORE BLANKS
231 000266 005001 MOV #DSTKTBL,R2 ; KEYWORD TABLE
232 000270 012702 000000' MOV #DSTSTBL,R5 ; STATE TABLE
233 000274 012705 000000' CALL TPARS ; PARSE THE REST OF THE LINE
234 000300 004767 000000G BCC 20$; NORMAL RETURN IF NO ERROR
235 000304 103012 MOV .ERROR,R0 ; CHECK FOR ERROR DETAIL TEXT
236 000306 016700 BEQ 15$; IF EQ, NO DETAIL TEXT
237 000312 001405 ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
238 000314 000314 MOV #-2,$ERROR ; SUPPRESS SYNTAX MESSAGE
239 000320 012767 177776 000000G 15$: INC $ERROR ; INDICATE SYNTAX ERROR
240 000326 005267 000000G 20$: MOV (SP)+,R5 ; RESTORE R5
241 000332 012605 RETURN
242 000334 000207
243
244 .SBTTL SETTYP - SET DESTINATION TYPE
245 .SBTTL SETNAM - SET DST NAME
246 .SBTTL SETPRI - SET PRIORITY
247 .SBTTL SETOBJ - SET OBJECT NUMBER
248 .SBTTL SETTSK - SET TASK NAME
249
250 +
251 ** SETTYP - SET DESTINATION TYPE
252 ** SETNAM - SET DST NAME
253 ** SETPRI - SET PRIORITY
254 ** SETOBJ - SET OBJECT NUMBER
255 ** SETTSK - SET TASK NAME
256
257 INPUTS .TPARS VARIABLES.
258
259 OUTPUT VALUE SET IN TEMPLATE.
260
261 -
262
263 000336 SETTYP: $GTR50 #C.TYP,#1

```

```

243 .SBTTL $TMDTE - BUILD DTE TEMPLATE
244
245 *** - $TMDTE - BUILD DTE TEMPLATE
246
247 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
248
249 INPUT:
250 R3-R5 - TPARS REGISTERS
251
252 OUTPUT:
253 C-BIT = SUCCESS/FAILURE
254 THE CHANNEL TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
255
256
257
258 $TMDTE::
259 MOV R5, -(SP) ; SAVE R5
260 MOV #C.LEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
261 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
262 MOV RO, TEMP ; SAVE TEMPLATE ADDRESS
263 TST (R0)+ ; POINT TO C.STS
264 MOV RO, .FLAGS ; SAVE FOR OPTION SETTING
265 MOV #CS.DTE, (R0) ; INDICATE THAT THIS IS A DTE$DF TEMPLATE
266 CLR .ERROR ; NO ERROR DETAIL YET
267 CLR STA ; ASSUME STATE = OFF
268 ADD #L$LENX, $BYTMN ; UPDATE MINIMUM ALLOCATION
269 CLR R1 ; IGNORE BLANKS
270 MOV #DTEKTB, R2 ; KEYWORD TABLE
271 MOV #DTESTB, R5 ; STATE TABLE
272 CALL TPARS ; PARSE THE REST OF THE LINE
273 BCC 20$; NORMAL RETURN IF NO ERROR
274 MOV .ERROR, R0 ; CHECK FOR ERROR DETAIL TEXT
275 BEQ 15$; IF EQ, NO DETAIL TEXT
276 ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
277 MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
278 INC $ERROR ; INDICATE SYNTAX ERROR
279 MOV (SP)+, R5 ; RESTORE R5
280 RETURN
281
258 000264
259 000264 010546
260 000266 012701 000040
261 000272 004767 000000G
262 000276 010067 000000G
263 000302 005720
264 000304 010067 000000G
265 000310 012710 000000G
266 000314 005067 000000G
267 000320 005067 177736
268 000324 062767 000000G 000000G
269 000332 005001
270 000334 012702 000000'
271 000340 012705 000000'
272 000344 004767 000000G
273 000350 103012
274 000352 016700 000000G
275 000356 001405
276 000360
277 000364 012767 177776 000000G
278 000372 005267 000000G 15$:
279 000376 012605 20$:
280 000400 000207
281

```



```

145 .SBTTL $TMEVT - SETUP EVENT TEMPLATE
146 :+
147 *** - $TMEVT - SETUP EVENT TEMPLATE
148 :
149 INPUT:
150 R3-R5 - TPARS REGISTERS
151 :
152 OUTPUT:
153 C-BIT = SUCCESS/FAILURE
154 IF SUCCESS, THE OBJECT TEMPLATE IS STORED IN THE END OF TASK BUFFER.
155 :
156 :-
157
158 000004 $TMEVT::
159 000004 010546 MOV R5,-(SP) ; SAVE R5
160 000006 012701 000020 MOV #C.LEN,R1 ; GET LENGTH OF ALLOCATION
161 000012 004767 000000G CALL $T1ALC ; TRY TO ALLOCATE CORE BLOCK
162 000016 010067 177756 MOV R0,TEMP ; STORE TEMPLATE ADDRESS
163 000022 012760 000000G 000000G MOV #CS.EVT,C.STS(R0) ; INDICATE THAT THIS IS AN OBJ$DF TEMPLATE
164 000030 005060 000006 CLR C.EVT+0(R0) ; INITIALIZE EVENT MASKS
165 000034 005060 000010 CLR C.EVT+2(R0) ; ...
166 000040 005060 000012 CLR C.EVT+4(R0) ; ...
167 000044 005060 000014 CLR C.EVT+6(R0) ; ...
168 000050 005001 CLR R1 ; IGNORE BLANKS
169 000052 012702 000000' MOV #EVT$TB,R2 ; KEYWORD TABLE
170 000056 012705 000000' MOV #EVT$TB,R5 ; STATE TABLE
171 000062 004767 000000G CALL ,TPARS ; PARSE THE REST OF THE LINE
172 000066 103002 BCC 20$; NORMAL RETURN IF NO ERROR
173 000070 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
174 000074 016700 177700 20$: MOV TEMP,R0 ; RETRIEVE TEMPLATE ADDRESS
175 000100 004767 000016 CALL FINEVT ; FIND PLACE FOR THE OBJECT DEFINITION
176 000104 004767 000000G CALL $QINSN ; INSERT ENTRY INTO QUEUE
177 000110 062767 000000G 000000G ADD #F.LENX,$BYTMN ; ADJUST MINIMUM EXTENSION
178 000116 012605 MOV (SP)+,R5 ; RESTORE R5
179 000120 000207 RETURN

```

```

103 000016 .ASECT
104 000000 . = 0
105 000000 F.LNK: .BLKW 1 ; Link word
106 000002 F.CNT: .BLKW 1 ; Use count
107 000004 .BLKB 1 ; Filespec string length
108 000005 F.DEV: .BLKB MXFSP ; Filespec string (ASCII)
109 000043 F.LEN = .PSECT
110 000000
111 ;
112 ; Local data
113
114 000000 000000 FILE: .WORD 0 ; Offset of current file in template
115 000002 FD: .BLKB F.LEN ; Prototype file descriptor
116
117 ;
118 ; SYNTAX ERROR DETAIL STRINGS
119 ;
120 .NLIST BEX
121 EX$SEV = 4
122 000045 004 103 106 ERRFSP: .ASCIZ <EX$SEV>/CFE -- Error in filespec/
123 000077 004 103 106 ERRUIC: .ASCIZ <EX$SEV>/CFE -- Error in UIC/
124 .EVEN
125 .LIST BEX
126

```

TMPFEA - PARSE FEATURES DEFINIT MACRO V05.03b Saturday 29-Jun-85 05:35  
Table of contents

|     |     |                                             |
|-----|-----|---------------------------------------------|
| 6-  | 129 | \$TMFEA - SETUP FEATURES TEMPLATE           |
| 7-  | 157 | SETDV1 - SETUP 1ST CHARACTER OF DEVICE NAME |
| 8-  | 175 | SETDV2 - SETUP 2ND CHARACTER OF DEVICE NAME |
| 9-  | 193 | SETWD1 - SETUP INITIAL FEATURES WORD        |
| 10- | 218 | SETWD2 - SETUP THE SETTABLE STATES WORD     |
| 11- | 240 | SETWD3 - SETUP THE CURRENT STATES WORD      |

[illegible][illegible]

```

351
352
353
354
355
356
357
358
359
360
361
362
363
364
365 000356
366 000356 016700 177420
367 000362 116760 000000G 000015
368 000370 026727 000000G 000200
369 000376 103402
370 000400 062716 000002
371 000404 000207

.SBTTL SETEXT - SETUP THE LLC PROCESS EXTENSION
:
: +
: *** - SETEXT - SETUP THE LLC PROCESS EXTENSION
:
: INPUTS:
: TEMP = THE TEMPLATE ADDRESS
: .PNUMB = THE SIZE OF THE PROCESS EXTENSION IN BLOCKS
:
: OUTPUTS:
: THE PROCESS EXTENSION FIELD IS MOVED INTO C.EXT OF THE TEMPLATE
: TRANSITION IS ACCEPTED/REJECTED
:
: -
:
SETEXT:
10V TEMP,R0 ; GET THE TEMPLATE ADDRESS
MOVB .PNUMB,C.EXT(R0) ; STORE THE PROCESS EXTENSION
CMP .PNUMB,#128. ; IS THE SIZE IN RANGE ?
BLO 10$; IF LO, YES, . OKAY
ADD #2,(SP) ; ELSE, REJECT TRANSITION
10$: RETURN

```

```

141 .SBTTL SETSTA - SET UP LOGGING STATE
142
143 ;+
144 *** - SETSTA - SET UP LOGGING STATE
145 ;
146 THIS ACTION ROUTINE SETS THE LOGGING STATE.
147 ;
148 INPUTS:
149 .PNUMB - LOGGING STATE
150 ;
151 OUTPUTS:
152 C.STA - LOGGING STATE
153 ;
154 -
155 SETSTA: MOV TEMP,R0 ; RETRIEVE POINTER TO TEMPLATE
156 TST .PNUMH ; MUST BE BYTE VALUE
157 BNE 10$; BR IF NOT BYTE VALUE
158 TSTB .PNUMB+1 ;
159 BNE 10$; BR IF NOT BYTE VALUE
160 MOVB .PNUMB,R1 ; GET LOGGING STATE
161 BLT 10$; BR IF INVALID LOGGING STATE
162 CMP R1,#MAXSTA ; VALID LOGGING STATE?
163 BHI 10$; BR IF NO
164 MOVB R1,C.STA(R0) ; STORE LOGGING STATE
165 BR 20$; FINISH UP
166 ADD #2,(SP) ; REJECT THE TRANSITION
167 RETURN
168
169 .END
10$:
20$:
000056 016700 177716
000062 005767 000000G
000066 001014
000070 105767 000001G
000074 001011
000076 116701 000000G
000102 002406
000104 020127 000007
000110 101003
000112 110160 000004
000116 000402
000120 062716 000002
000124 000207
000001

```

274  
 275  
 276  
 277  
 278  
 279  
 280  
 281  
 282  
 283  
 284  
 285  
 286  
 287  
 288  
 289  
 290  
 291  
 292  
 293  
 294  
 295  
 296

000312  
 000312 016700 177462  
 000316 016701 000000G  
 000322 001410  
 000324 020127 001777  
 000330 101005  
 000332 050160 000052  
 000336 050167 000000G  
 000342 000402  
 000344 062716 000002  
 000350 000207

```

;+
;*** - SETNNM - SET NODE NUMBER
;
;INPUT:
; .PNUMB = NODE NUMBER
; .PNUMH = NODE NUMBER (HIGH ORDER)
;
;OUTPUT:
; NUMBER IS STORED IN C.NUM OF TEMPLATE
;-
;
SETNNM: MOV TEMP,R0 ; POINT R0 AT TEMPLATE
 MOV .PNUMB,R1 ; PICK UP PARSED VALUE
 BEQ 5$; IF EQ, ILLEGAL NODE NUMBER
 CMP R1,#1023. ; IS THE NODE NUMBER IN RANGE ?
 BHI 5$; IF HI, OUT OF RANGE
 BIS R1,C.NUM(R0) ; STORE THE NODE NUMBER
 BIS R1,$NTADD ; ..TWICE
 BR 10$; OK
5$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
10$: RETURN

```

```

204 .SBTTL SETYPE - SETUP OBJECT TYPE
205 +
206 *** - SETYPE - SETUP OBJECT TYPE
207 :
208 INPUT:
209 .PNUMB = THE OBJECT NUMBER
210 .PNUMH = THE OBJECT NUMBER (HIGH ORDER)
211 TEMP = TEMPLATE ADDRESS
212 :
213 OUTPUT:
214 THE OBJECT NUMBER IS MOVED INTO C.TYP OF THE TEMPLATE
215 C-BIT = SUCCESS/FAILURE
216 :
217 -
218
219 000152
220 000152 016700 177622 SETYPE: MOV TEMP,R0 ; POINT AT TEMPLATE
221 000156 116760 000000G 000004 MOVB .PNUMB,C.TYP(R0) ; STORE THE OBJECT TYPE
222 000164 005767 000000G TST .PNUMH ; IS THIS A WORD VALUE ?
223 000170 001003 BNE 10$; IF NE, NO .. REJECT
224 000172 105767 000001G TSTB .PNUMB+1 ; IS THIS A BYTE VALUE ?
225 000176 001402 BFC 20$; IF EQ, YES LEAVE
226 000200 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT TRANSITION
227 000204 000207 20$: RETURN

```



150 000004  
151  
152 000004

TRAN\$ \$EOS,\$EXIT,SETFLG  
STATE\$

```

; SBTL MACRO DEFINITIONS
; LOCAL MACROS

.MACRO PRINT TEXT
MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
CALL $PRINT ; PRINT MESSAGE
.ENDM

.MACRO ERROR$ TEXT
; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
; MESSAGE STRING.
; IF DIF <TEXT><R0>
MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
.ENDC
CALL $CFERR ; PRINT ERROR MESSAGE
.ENDM

.MACRO EPRINT TEXT
; PRINT TEXT ON ERROR LUN
; IF DIF <TEXT><R0>
MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
.ENDC
CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
.ENDM EPRINT

; GET NUMBER WITH RANGE CHECKING
.MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
MOV OFFS,R2
MOV LO,R0
MOV HI,R1
; IF NB <OPT>
BIS OPT,@.FLAGS
.ENDC
JMP $GTNUM
.ENDM

.MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
MOV OFFS,R2
MOV LO,R0
MOV HI,R1
; IF NB <OPT>
BIS OPT,@.FLAGS
.ENDC
JMP $GTNMB
.ENDM

; GET RAD50 WITH FIELD SIZE CHECKING
.MACRO $GTR50 OFFS,LEN,OPT
MOV OFFS,R2

```

.TITLE TMPPVC - PARSE PERMANENT VIRTUAL CIRCUIT MACRO  
.IDENT /V05.00/  
.ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE PERMANENT VIRTUAL CIRCUIT DESCRIPTOR

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-AUG-81  
DECNET-11M/S V3.1  
DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

```

LL SSSSSSSS TTTT'TTTTTT
LL SSSSSSSS TTTT'TTTTTT
 SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLLL SSSSSSSS TT
LLLLLLLLLLL SSSSSSSS TT

```

```

140 .SBTTL TPARS STATE TABLES
141
142 ; INITIALIZE TABLES
143 ;
144
145 ISTAT$ DSASTB, DSAKTB
146
147 ; PROCESS 'DSA$DF'
148 ;
149 ; ALL PARAMETERS OPTIONAL
150 000210 STATES$; REMOTE DTE NUMBER
151 TRANS$ $EOS,$EXIT
152 000210 TRANS$ $LAMDA,, $CLERR, ERRNUM, .ERROR
153 000210 STATES$
154 TRANS$ 'DTE, DSA1, SETNUM
155 000210 TRANS$ $LAMDA
156
157 000210 STATES$ DSA1 ; SUBADDRESS LOW RANGE
158 TRANS$ $EOS,$EXIT
159 000210 TRANS$ <'>,, $CLERR, ERRSLD, .ERROR
160 000210 STATES$
161 TRANS$ $NUMBR, DSA2, SETSLD
162 000210 TRANS$ $LAMDA
163
164 000210 STATES$ DSA2 ; SUBADDRESS HI RANGE
165 TRANS$ $EOS,$EXIT
166 000210 TRANS$ <'>,, $CLERR, ERRSHI, .ERRDR
167 000210 STATES$
168 TRANS$ $NUMBR, DSA3, SETSHI
169 000210 TRANS$ $LAMDA
170
171 000210 STATES$ DSA3 ; USER GROUP
172 TRANS$ $EOS,$EXIT
173 000210 TRANS$ <'>,, $CLERR, ERRCUG, .ERROR
174 000210 STATES$
175 TRANS$ $STRNG,, SETCUG
176
177 000210 STATES$
178 TRANS$ $EOS, $EXIT
179
180 ; SUBEXPRESSION TO PARSE DTE ADDRESS STRING
181 ;
182
183 000210 STATES$ DTE
184 TRANS$ $DIGIT, ADR1
185 000210 TRANS$ <'*>
186 000210 STATES$ ADR1
187 TRANS$ $DIGIT, ADR1
188 000210 TRANS$ <'*> ADR1
189 000210 TRANS$ $LAMDA, $EXIT
190
191 ; FINAL STATE
192 ;
193 000210 STATES$
194

```

```

186 .SBTTL $TMDSC - BUILD DSC TEMPLATE
187 +
188 *** - $TMDSC - BUILD DSC TEMPLATE
189 :
190 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
191 :
192 INPUT:
193 R3-R5 - TPARS REGISTERS
194 :
195 OUTPUT:
196 C-BIT = SUCCESS/FAILURE
197 THE DSC TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
198 :
199 -
200
201 $TMDSC::
202 MOV R5, -(SP) ; SAVE R5
203 MOV #D$CLEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
204 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
205 MOV R0, .TEMP ; SAVE TEMPLATE ADDRESS
206 TST (R0)+ ; POINT TO C.STS
207 MOV R0, .FLAGS ; SAVE FOR OPTION SETTING
208 MOV #CS,DSC,(R0) ; INDICATE THAT THIS IS A DSC$DF TEMPLATE
209 CLR .ERROR ; NO ERROR DETAIL YET
210 CLR R1 ; IGNORE BLANKS
211 MOV #D$CKTB, R2 ; KEYWORD TABLE
212 MOV #D$CSTB, R5 ; STATE TABLE
213 CALL .TPARS ; PARSE THE REST OF THE LINE
214 BCS 10$; IF CS, ERROR DURING PARSE
215 MOV .TEMP, R1 ; POINT TO TEMPLATE
216 CMPB D$CVLN(R1), D$CMLN(R1) ; COMPARE LENGTHS
217 BEQ 20$; IF EQ, OK
218 MOV #ERRLEN, .ERROR ; ELSE LENGTH ERROR
219 BCC 20$; NORMAL RETURN IF NO ERROR
220 MOV .ERROR, R0 ; CHECK FOR ERROR DETAIL TEXT
221 BEQ 15$; IF EQ, NO DETAIL TEXT
222 ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
223 MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
224 INC $ERROR ; INDICATE SYNTAX ERROR
225 MOV (R0)+, R5 ; RESTORE R5
226 RETURN

```

TMPOST - PARSE DESTINATION DESC MACRO V05.03b Saturday 29-Jun-85 05:33 <sup>N 3</sup> Page 7-1  
SETTSK - SET TASK NAME

|     |        |                 |                   |
|-----|--------|-----------------|-------------------|
| 261 | 000352 | SETNAM: \$GTR50 | #C.NAM,#2         |
| 262 |        |                 |                   |
| 263 | 000366 | SETPRI: \$GTNMB | #C.PRI,,#255.     |
| 264 |        |                 |                   |
| 265 | 000406 | SETOBJ: \$GTNMB | #C.OBJ,,#255.     |
| 266 |        |                 |                   |
| 267 | 000426 | SETTSK: \$GTR50 | #C.TSK,#2,#CS.TSK |
| 268 |        |                 |                   |
| 269 | 000001 |                 | .END              |

TMPOST - PARSE DESTINATION DESC MACRO V05.03b Saturday 29-Jun-85 05:33 <sup>N 3</sup> Page 7-2  
Symbol table

\$ENDPDB= \*\*\*\*\* GY \$TALOC= \*\*\*\*\* GY

```

283 .SBTTL SETADR - SET DTE ADDRESS IN TEMPLATE
284 .SBTTL SETHSH - SET HASH TABLE SIZE IN DTE
285 .SBTTL SETCT - SET COUNTER TIMER VALUE
286 .SBTTL SETNET - SET NETWORK NAME IN DTE TEMPLATE
287
288 ;+
289 ; ** SETADR - SET DTE ADDRESS IN TEMPLATE
290 ; ** SETHSH - SET HASH TABLE SIZE IN DTE
291 ; ** SETCT - SET COUNTER TIMER VALUE
292 ; ** SETNET - SET NETWORK NAME IN DTE TEMPLATE
293
294 ; INPUTS: .TPARS CONVENTIONS.
295
296 ; OUTPUT: VALUE SET IN TEMPLATE.
297 ; -
298 SETADR: $GTR50 #C.ADR,#5
299
300 000416 012700 000020 SETHSH: MOV #16.,R0 ; TEST FOR POWER OF TWO
301 000422 006300 10$: ASL R0 ; X 2
302 000424 020067 000000G CMP R0,.PNUMB ; CHECK PARSED NUMBER
303 000430 002774 BLT 10$; IF LT, CONTINUE
304 000432 003020 BGT REJECT ; IF GT, REJECT TRANSITION
305 000434 006300 ASL R0 ; X 4
306 000436 006300 ASL R0 ; IN BYTES
307 000440 062700 000003 ADD #3,R0 ; ROUND UP
308 000444 042700 000003 BIC #3,R0 ;
309 000450 060067 000000G ADD R0,$BYTMN ; UPDATE MINIMUM ALLOCATION
310 000454 $GTNUM #C.HSH,#32.,#512.
311
312 000474 062716 000002 REJECT: ADD #2,(SP) ; REJECT TRANSITION
313 000500 000207 RETURN
314
315 000502 SETCT: $GTNUM #C.CT,#0,#-1
316
317 000522 SETNET: $GTR50 #C.NET,#2,#CS.NET
318
319 .SBTTL SETSTA - SET DTE STATE IN TEMPLATE
320
321 ;+
322 ; ** SETSTA - SET DTE STATE IN TEMPLATE
323 ;
324 ; INPUTS: .TPARS CONVENTIONS.
325
326 ; OUTPUT: DTE STATE SET IN DTE TEMPLATE.
327 ; -
328
329 000544 016700 000000G SETSTA: MOV .TEMP,R0 ; POINT TO TEMPLATE
330 000550 016760 177506 000026 MOV STA,C.STA(R0) ; LOAD STATE INTO TEMPLATE
331 000556 000207 RETURN
332
333 .SBTTL SETDEV - SET DEVICE NAME IN LINE ID
334 .SBTTL SETCON - SET CONTROLLER NUMBER IN LINE ID
335 .SBTTL SETUNI - SET UNIT NUMBER IN LINE ID
336 .SBTTL SETTRB - SET TRIB NUMBER IN LINE ID
337
338 ;+
339 ; ** SETDEV - SET DEVICE NAME IN LINE ID
340 ; ** SETCON - SET CONTROLLER NUMBER IN LINE ID

```



```

181 .SBTTL FINEVT - FIND NEXT EVENT IN SORTED LIST
182 ;*
183 ;*** - FINEVT - FIND NEXT EVENT IN SORTED LIST
184 ;
185 ; INPUTS:
186 ; R0 = TEMPLATE TO BE INSERTED
187 ;
188 ; OUTPUTS:
189 ; R1 = ADDRESS OF TEMPLATE TO PRECEED CURRENT TEMPLATE
190 ;
191 ; -
192
193 FINEVT:
194 000122 012702 000000G MOV #$TMLST,R2 ; GET START ADDRESS OF TEMPLATES
195 000126 010201 MOV R2,R1 ; SAVE ADDRESS OF PREVIOUS
196 000130 011202 MOV (R2),R2 ; GET NEXT TEMPLATE IN LIST
197 000132 001414 BEQ 20$; IF EQ, END OF LIST
198 000134 022762 000000G 000000G CMP #CS.EVT,C.STS(R2) ; IS THIS AN EVENT TEMPLATE?
199 000142 001371 BNE 10$; IF NE, NO .. KEEP LOOKING
200 000144 026062 000004 000004 CMP C.CLS(R0),C.CLS(R2) ; ELSE, DOES TEMPLATE GO HERE?
201 000152 101365 BHI 10$; IF HI, NO .. KEEP LOOKING
202 000154 026062 000016 000016 CMP C.SNK(R0),C.SNK(R2) ; ELSE, DOES TEMPLATE GO HERE?
203 000162 101361 BHI 10$; IF HI, NO .. KEEP LOOKING
204 000164 000207 20$: RETURN ; ELSE, OKAY

```

```

128 .SBTTL State tables
129
130 000054 COMMA = '<','>'
131
132 000124 ISTAT$ FILSTB,FILKTB
133
134 000124 STATES ; Optional secondary loader file name
135 000124 TRANS $LAMDA,, CLEFIL,F$SEC, FILE
136 000124 STATES
137 000124 TRANS $EOS, $EXIT
138 000124 TRANS COMMA, S10
139 000124 TRANS !FILSPC,, STFIL
140
141 000124 STATES S10 ; Optional tertiary loader file name
142 000124 TRANS $LAMDA,, CLEFIL,F$TER, FILE
143 000124 STATES
144 000124 TRANS $EOS, $EXIT
145 000124 TRANS COMMA, S20
146 000124 TRANS !FILSPC,, STFIL
147
148 000124 STATES S20 ; Load file name
149 000124 TRANS $LAMDA,, CLEFIL,F$LD, FILE
150 000124 STATES
151 000124 TRANS $EOS, $EXIT
152 000124 TRANS COMMA, S30
153 000124 TRANS !FILSPC,, STFIL
154
155 000124 STATES S30 ; Diagnostics file name
156 000124 TRANS $LAMDA,, CLEFIL,F$DIA, FILE
157 000124 STATES
158 000124 TRANS $EOS, $EXIT
159 000124 TRANS COMMA, S40
160 000124 TRANS !FILSPC,, STFIL
161
162 000124 STATES S40 ; Dump file name
163 000124 TRANS $LAMDA,, CLEFIL,F$DUM, FILE
164 000124 STATES
165 000124 TRANS $EOS, $EXIT
166 000124 TRANS !FILSPC,$EXIT, STFIL
167
168 ;
169 ; Subexpression to parse filespec
170 ;
171 000124 STATES FILSPC
172 000124 TRANS !STRING,$EXIT, CKFSP
173 000124 TRANS $EXIT
174
175 ; Subexpression to collect string of any type of characters
176 ;
177 000124 STATES STRING
178 000124 TRANS !GETUIC,GTRST, CKUIC
179 000124 TRANS $EXIT
180
181 000124 STATES GTRST
182 000124 TRANS COMMA, $EXIT
183 000124 TRANS $EOS, $EXIT
184 000124 TRANS $ANY, STRING

```

.TITLE TMPFEA - PARSE FEATURES DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE FEATURES DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/s V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

TMPLLC - CFE PARSE LLC PROCESS MACRO V05.03b Saturday 29-Jun-85 M 8 05:35  
Table of contents

|     |     |                                                          |
|-----|-----|----------------------------------------------------------|
| 6-  | 173 | \$TMLLC - SETUP LLC TEMPLATE                             |
| 7-  | 205 | SETNAM - SET UP PROCESS NAME                             |
| 8-  | 231 | SETFLG - SETUP THE FLAGS WORD                            |
| 9-  | 252 | SETPRI - SET THE PRIORITY                                |
| 10- | 278 | SETLIN - SET LINE NUMBER                                 |
| 11- | 306 | SETDEV - STORE THE PSEUDO DEVICE NAME                    |
| 12- | 329 | SETNLT - SET NUMBER OF LINE TABLES TO ALLOCATE SPACE FOR |
| 13- | 351 | SETEXT - SETUP THE LLC PROCESS EXTENSION                 |
| 14- | 373 | SETCTM - COUNTER TIMER                                   |

```

373 .SBTTL SETCTM - COUNTER TIMER
374 :+
375 :*** - SETCTM - SETUP COUNTER TIMER
376 :THIS ACTION ROUTINE SETS UP THE COUNTER TIMER FOR THIS PROCESS
377 :
378 :INPUT:
379 :.PNUMB = COUNTER TIMER
380 :.PNUMH = (HIGH ORDER)
381 :
382 :OUTPUT:
383 :THE COUNTER TIMER IS STORED IN C.CTM OF THE TEMPLATE
384 :-
385 :
386 :
387 :
388 000406 SETCTM: MOV TEMP,RO ; GET START OF TEMPLATE
389 000406 016700 MOV .PNUMB,C.CTM(RO) ; STOP THE COUNTER TIMER
390 000412 016760 000000G 00C016 TST .PNUMH ; IS IT ONLY A WORD VALUE ?
391 000420 005767 000000G BNE 10$; IF NE NO - REJECT
392 000424 001003 TSTB .PNUMB+1 ; IS IT A BYTE VALUE ?
393 000426 105767 000001G BEQ 20$; IF EQ, YES - OKAY
394 000432 001402 10$: ADD #2,(SP) ; ELSE, REJECT TRANSITION
395 000434 062716 000002 20$: RETURN
396 000440 000207
397
398 000001 .END

```

TMPLOG - CFE PARSE LOGGING STAT MACRO V05.03b Saturday 29-Jun-85 05:36 Page 8-1

## Symbol table

|                  |                    |                     |                   |                   |
|------------------|--------------------|---------------------|-------------------|-------------------|
| CS.LOG= ***** GX | LOGSTB 000000RG    | 002 \$BLANK= 000006 | \$LAMDA= 000000   | \$\$\$FLG= 177777 |
| C.LEN 000005     | MAXSTA= 000007     | \$DIGIT= 000024     | \$NUMBR= 000002   | \$\$\$KEY= 177777 |
| C.STA 000004     | RA = 000076        | \$DNUMB= 000014     | \$RAD50= 000016   | \$\$\$STA= 000000 |
| C.STS = ***** GX | SETSTA 000056R     | \$EOS = 000012      | \$STRNG= 000004   | .PNUMB= ***** GX  |
| EXIT 000004R     | 002 TEMP 000000R   | \$ERROR= ***** GX   | \$SUBXP= 000010   | .PNUMH= ***** GX  |
| LA = 000074      | \$ALPHA= 000022    | \$EXIT = 000000     | \$TALOC= ***** GX | .TPARS= ***** GX  |
| LOGKTB 000000RG  | 003 \$ANY = 000020 | \$FAIL = 177777     | \$TMLG 000002RG   |                   |

|                |     |                    |
|----------------|-----|--------------------|
| . ABS. 000005  | 000 | (RW,I,GBL,ABS,OVR) |
| 000126         | 001 | (RW,I,LCL,REL,CON) |
| \$STATE 000010 | 002 | (RW,D,LCL,REL,CON) |
| \$KTAB 000000  | 003 | (RW,D,LCL,REL,CON) |
| \$KSTR 000000  | 004 | (RW,D,LCL,REL,CON) |

Errors detected: 0

## \*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10406 Words ( 41 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:08.57

SY:TMPLOG.V2,[132,134]TMPLOG/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPLOG

```

298
299
300
301
302
303
304
305
306
307
308
309
310 000352
311 000352 016700 177422
312 000356 016701 000000G
313 000362 001414
314 000364 020127 000077
315 000370 101011
316 000372 000301
317 000374 006301
318 000376 006301
319 000400 010160 000054
320 000404 052760 000000G 000000G
321 000412 000402
322 000414 062716 000002
323 000420 000207

;+
; *** - SETHOA - SET HOST AREA
;
; INPUT:
; .PNUMB = NODE NUMBER
; .PNUMH = NODE NUMBER (HIGH ORDER)
;
; OUTPUT:
; NUMBER IS STORED IN C.HOS OF TEMPLATE
;
;-

SETHOA:
MOV TEMP,R0 ; POINT RO AT TEMPLATE
MOV .PNUMB,R1 ; PICK UP AREA NUMBER
BEQ 5$; BR IF TOO LOW
CMP R1,#63. ; TOO HIGH ?
BHI 5$; IF HI, ERROR
SWAB R1 ; DIDDLE WITH FORMAT
ASL R1 ; ...
ASL R1 ; ...
MOV R1,C.HOS(R0) ; STORE THE NODE NUMBER
BIS #NO$HOA,C.STS(R0) ; REMEMBER WE SAW AN AREA
BR 10$
5$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
10$: RETURN

```

TMPOBJ - CFE PARSE OBJECT DEFIN MACRO V05.03b Saturday 29-Jun-85 05:37 Page 9  
 SETNAM - SET UP OBJECT NAME

```

229 .SBTTL SETNAM - SET UP OBJECT NAME
230 ; *
231 ; *** - SETNAM - SETUP OBJECT NAME
232 ;
233 ; THIS ACTION ROUTINE SETS UP THE OBJECT NAME.
234 ;
235 ; INPUT:
236 ; .PSTPT = START ADDRESS OF THE OBJECT NAME IN ASCII
237 ; .PSTCN = NUMBER OF CHARACTERS IN THE OBJECT NAME
238 ; TEMP = TEMPLATE ADDRESS
239 ;
240 ; OUTPUT:
241 ; C.NAM = RAD50 OBJECT NAME (2 WORDS)
242 ;
243 ; -
244
245 SETNAM:
246 000206 016700 000000G MOV .PSTPT,R0 ; GET ADDRESS OF ASCII OBJECT NAME
247 000212 010701 MOV PC,R1 ; PERIODS ARE ACCEPTABLE
248 000214 004767 CALL $CAT5 ; CONVERT OBJECT NAME TO RAD50
249 000220 016702 177554 MOV TEMP,R2 ; GET START OF TEMPLATE
250 000224 010162 000006 MOV R1,C.NAM(R2) ; STORE OBJECT NAME IN TEMPLATE
251 000230 010701 MOV PC,R1 ; PERIODS ARE ACCEPTABLE
252 000232 004767 000000G CALL $CAT5 ; CONVERT THE REST OF THE OBJECT NAME
253 000236 016702 177536 MOV TEMP,R2 ; GET START OF TEMPLATE .. AGAIN !!
254 000242 010162 000010 MOV R1,C.NAM+2(R2) ; STORE THE SECOND WORD
255 000246 022767 000006 000000G CMP #6,.PSTCN ; IS THE OBJECT NAME TOO BIG ?
256 000254 103002 BHIS 10$; IF HIS, NO
257 000256 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
258 000262 000207 10$: RETURN

```

TMPOBJ - CFE PARSE OBJECT DEFIN MACRO V05.03b Saturday 29-Jun-85 05:37 Page 10  
 SETVFY - SETUP OBJECT VERIFICATION LEVEL



TMPPAR - CFE PARSE PARTITION DE MACRO V05.03b Saturday 29-Jun-85 05:38 Page 6  
 \$TMPAR - SETUP PARTITION TEMPLATE

```

154 .SBTTL $TMPAR - SETUP PARTITION TEMPLATE
155 ;+
156 *** - $TMPAR - SETUP PARTITION TEMPLATE
157 ;
158 INPUT:
159 R3-R5 - TPARS REGISTERS
160 ;
161 OUTPUT:
162 C-BIT = SUCCESS/FAILURE
163 IF SUCCESS, THE PARTITION TEMPLATE IS STORED IN THE END OF TASK BUFFER.
164 ;
165 :-
166
167 000004 $TMPAR::
168 000004 010546 MOV R5,-(SP) ; SAVE R5
169 000006 012701 000020 MOV #C.LEN,R1 ; GET LENGTH OF ALLOCATION
170 000012 004767 000000G CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
171 000016 010067 177760 MOV RO,TEMP ; STORE TEMPLATE ADDRESS
172 000022 010001 MOV RO,R1 ; ..TWICE
173 000024 062701 000004 ADD #C.NAM0,R1 ; POINT AT PARTITION NAMES
174 000030 012702 000004 MOV #4,R2 ; REPEAT COUNT
175 000034 005021 CLR (R1)+ ; CLEAR OUT 2 PARTITION NAMES
176 000036 005302 DEC R2 ; ...
177 000040 003375 BGT 10$;
178 000042 012760 000000G 000000G MOV #CS.PAR,C.STS(R0) ; INDICATE THAT THIS IS A PAR$DF TEMPLATE
179 000050 005060 000016 CLR C.ALL(R0) ; ASSUME POOL BYTE-AREA EXTENSION NOT PRESENT
180 000054 005001 CLR R1 ; IGNORE BLANKS
181 000056 012702 000000' MOV #PARKTB,R2 ; KEYWORD TABLE
182 000062 012705 000000' MOV #PARSTB,R5 ; STATE TABLE
183 000066 004767 000000G CALL .TPARS ; PARSE THE REST OF THE LINE
184 000072 103002 BCC 20$; NORMAL RETURN IF NO ERROR
185 000074 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
186 000100 012605 MOV (SP)+,R5 ; RESTORE R5
187 000102 000207 RETURN

```

TMPPAR - CFE PARSE PARTITION DE MACRO V05.03b Saturday 29-Jun-85 05:38 Page 7  
 SETNMO - SET UP PARTITION NAME 0

```

114 MOV LEN,R1
115 .IF NB <OPT>
116 BIS OPT,@.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 ; MACRO CALLS
122 ;
123 ;
124 .MCALL ISTAT$,STAT$,TRANS$,CALLR

```

```

57 .SBTTL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61
62 .MACRO PRINT TEXT
63 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM PRINT
66
67 .MACRO ERRORS$ TEXT
68
69 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
70 ; MESSAGE STRING.
71
72 .IF DIF <TEXT><R0>
73 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
74 .ENDC
75 CALL $CFERR ; PRINT ERROR MESSAGE
76 .ENDM ERRORS$
77
78 .MACRO EPRINT TEXT
79
80 ; PRINT TEXT ON ERROR LUN
81
82 .IF DIF <TEXT><R0>
83 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
84 .ENDC
85 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
86 .ENDM EPRINT
87
88 ; GET NUMBER WITH RANGE CHECKING
89
90 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
91 MOV OFFS,R2
92 MOV LO,R0
93 MOV HI,R1
94 .IF NB <OPT>
95 BIS OPT,@.FLAGS
96 .ENDC
97 JMP $GTNUM
98 .ENDM
99
100 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
101 MOV OFFS,R2
102 MOV LO,R0
103 MOV HI,R1
104 .IF NB <OPT>
105 BIS OPT,@.FLAGS
106 .ENDC
107 JMP $GTNMB
108 .ENDM
109
110 ; GET RAD50 WITH FIELD SIZE CHECKING
111
112 .MACRO $GTR50 OFFS,LEN,OPT
113 MOV CFFS,R2

```

IMPRDT - PARSE REMOTE DIE IN CE MACRO V05.03b Saturday 29-Jun-85 05:40 M 16  
Table of contents

|    |     |                              |
|----|-----|------------------------------|
| 4- | 57  | MACRO DEFINITIONS            |
| 5- | 127 | LOCAL DATA                   |
| 6- | 138 | TPARS STATE TABLES           |
| 7- | 174 | \$TMRDT - BUILD RDT TEMPLATE |
| 8- | 212 | SETNAM - SET NAME            |
| 8- | 213 | SETADR - SET ADDRESS         |

196  
197  
198  
199  
200  
201  
202  
203  
204  
205  
206  
207  
208  
209  
210  
211  
212  
213  
214  
215  
216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231

000210  
000210 010546  
000212 012701 000000G  
000216 004767 000000G  
000222 010067 000000G  
000226 005720  
000230 010067 000000G  
000234 012710 000000G  
000240 005067 000000G  
000244 005001  
000246 012702 000000\*  
000252 012705 000000\*  
000256 004767 000000G  
103012  
000264 016700 000000G  
000270 001405  
000272  
000276 012767 177776 000000G  
000304 005267 000000G  
000310 012605  
000312 000207

```
.SBTTL $TMDSA - BUILD DSA TEMPLATE
*** - $TMDSA - BUILD DSA TEMPLATE
THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
INPUT:
R3-R5 - TPARS REGISTERS
OUTPUT:
C-BIT = SUCCESS/FAILURE
THE DSA TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
$TMDSA::
MOV R5, -(SP) ; SAVE R5
MOV #D$ALEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
$TALOC ; TRY TO ALLOCATE A CORE BLOCK
MOV R0, TEMP ; SAVE TEMPLATE ADDRESS
TST (R0)+ ; POINT TO C.STS
MOV R0, .FLAGS ; SAVE FOR OPTION SETTING
MOV #CS.DSA, (R0) ; INDICATE THAT THIS IS A DSA$DF TEMPLATE
CLR ERROR ; NO ERROR DETAIL YET
CLR R1 ; IGNORE BLANKS
MOV #DSAKTB, R2 ; KEYWORD TABLE
MOV #DSASTB, R5 ; STATE TABLE
CALL TPARS ; PARSE THE REST OF THE LINE
BCC 20$; NORMAL RETURN IF NO ERROR
MOV ERROR, R0 ; CHECK FOR ERROR DETAIL TEXT
BEQ 15$; IF EQ, NO DETAIL TEXT
ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
INC $ERROR ; INDICATE SYNTAX ERROR
MOV (SP)+, R5 ; RESTORE R5
RETURN
```

```

228 .SBTTL SETMSK - SET CALL MASK
229 .SBTTL SETVAL - SET CALL VALUE
230
231 ;+
232 ;** SETMSK - SET CALL MASK
233 ;** SETVAL - SET CALL VALUE
234
235 ; INPUTS .TPARS VARIABLES
236
237 ; OUTPUT VALUE SET IN TEMPLATE.
238 ; -
239 000304 016700 000000G SETVAL: MOV .PSTCN,R0 ; GET PARSED LENGTH
240 000310 020027 000040 CMP R0,#32. ; CHECK FOR LENGTH = 33.
241 000314 003045 BGT REJECT ; IF GT, REJECT TRANSITION
242 000316 016702 000000G MOV .TEMP,R2 ; POINT TO TEMPLATE
243 000322 110062 000000G MOVB R0,D$CVLN(R2) ; STORE LENGTH
244 000326 062700 000003 ADD #3,R0 ; ROUND UP
245 000332 042700 000003 BIC #3,R0 ; ...
246 000336 060000 000000G .DD R0,$BYTMN ; INCLUDE IN MINIMUM BYTE ALLOCATION
247 000342 $GTR50 #D$CVAL,#11.,#CS.VAL
248
249 000364 016700 000000G SETMSK: MOV .PSTCN,R0 ; GET PARSED LENGTH
250 000370 020027 000040 CMP R0,#32. ; CHECK FOR LENGTH = 33.
251 000374 003015 BGT REJECT ; IF GT, REJECT TRANSITION
252 000376 016702 000000G MOV .TEMP,R2 ; POINT TO TEMPLATE
253 000402 110062 000000G MOVB R0,D$CMLN(R2) ; STORE LENGTH
254 000406 $GTR50 #D$CMSK,#11.,#CS.MSK
255
256 000430 062716 000002 REJECT: ADD #2,(SP) ; REJECT TRANSITION
257 000434 000207 RETURN
258
259 000001 .END

```

|                     |                   |                   |                   |                    |
|---------------------|-------------------|-------------------|-------------------|--------------------|
| CS.DST= ***** GX    | D\$FLNX= ***** GX | SETTYP 000336R    | \$ERROR= ***** GX | \$TALOC= ***** GX  |
| CS.TSK= ***** GX    | ERRNAM 000041R    | \$ALPHA= 00002?   | \$EXIT = 000000   | \$TMDST 000224RG   |
| C.LEN = 000020      | ERRPUBJ 000133R   | \$ANY = 000020    | \$FAIL = 177777   | \$\$\$FLG= 177777  |
| C.NAM 000006        | ERRPRI 000102R    | \$BLANK= 000006   | \$GTNMB= ***** GX | \$\$\$KEY= 000001  |
| C.OBJ 000013        | ERRTSK 000171R    | \$BYTMN= ***** GX | \$GTR50= ***** GX | \$\$\$STA= 000000  |
| C.PRI 000012        | ERRTYP 000000R    | \$CFERR= ***** GX | \$LAMDA= 000000   | \$\$\$TMP= 000004R |
| C.TSK 000014        | EX\$SEV= 000004   | \$CLERR= ***** GX | \$NUMBR= 000002   | .ERROR= ***** GX   |
| C.TYP 0004          | SETNAM 000352R    | \$DIGIT= 000024   | \$RAD50= 000016   | .FLAGS= ***** GX   |
| DSTKTB 00000RG 003  | SETOBJ 000406R    | \$DNUMB= 000014   | \$STRNG= 000004   | .TEMP = ***** GX   |
| DSTSTB 000000RG 002 | SETPRI 000366R    | \$EOS = 000012    | \$SUBXP= 000010   | .TPARS= ***** GX   |
| DST1 000024R 00?    | SETTSK 000426R    |                   |                   |                    |

004

. ABS. 000020 000 (RW,I,GBL,ABS,OVR)  
 000450 001 (RW,I,LCL,REL,CON)  
 \$STATE 000114 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000004 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000010 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 11000 Words ( 43 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:22.11  
 SY:TMPDEST.V2,[132,134]TMPDEST/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPDEST

TMPDTE - PARSE LOCAL DTE DESCR MACRO V05.03b Saturday 29-Jun-85 N 4 05:33 Page 8-1  
SETTRB - SET TRIB NUMBER IN LINE ID

```
340 ; ** SETUNI - SET UNIT NUMBER IN LINE ID
341 ; ** SETTRB - SET TRIB NUMBER IN LINE ID
342 ;
343 ; INPUTS: .TPARS CONVENTIONS.
344 ;
345 ; OUTPUT: VALUE SET IN TEMPLATE.
346 ;
347 ;
348 SETDEV: $GTR50 #C.LIN,#1
349
350 SETCON: $GTNUM #C.LIN+2
351
352 SETUNI: $GTNUM #C.LIN+4,,,#CS.UNI
353
354 SETTRB: $GTNUM #C.LIN+6,,,#CS.TRB
355
356 .END
```

000560  
000574  
000614  
000642  
000001



```

206 .SBTTL SETCLS - SETUP EVENT CLASS
207
208 *** - SETCLS - SETUP EVENT CLASS
209
210 INPUT:
211 .PNUMB = THE EVENT CLASS NUMBER
212 .PNUMH = THE OBJECT NUMBER (HIGH ORDER)
213 TEMP = TEMPLATE ADDRESS
214
215 OUTPUT:
216 THE EVENT CLASS NUMBER IS MOVED INTO C.EVT OF THE TEMPLATE
217 C-BIT = SUCCESS/FAILURE
218
219 -
220
221 SETCLS:
222 000166 016700 177606 MOV TEMP,R0 ; POINT AT TEMPLATE
223 000172 026727 000000G 077700 CMP .PNUMB,#MAXCLS ; VALID EVENT CLASS VALUE?
224 000200 101014 BHI 10$; BR IF NO
225 000202 016760 000000G 000004 MOV .PNUMB,C.CLS(R0) ; STORE THE EVENT CLASS
226 000210 005:67 000000G TST .PNUMH ; IS THIS A WORD VALUE ?
227 000214 001006 BNE 10$; IF NE, NO
228 000216 010067 177560 MOV R0,STRADD ; POINT AT TEMPLATE
229 000222 062767 000006 177552 ADD #C.EVT,STRADD ; POINT AT FIRST EVENT MASK
230 000230 000402 BR 20$; FINISH UP
231 000232 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT TRANSITION
232 000236 000207 20$: RETURN

```

|     |        |        |                 |
|-----|--------|--------|-----------------|
| 185 |        |        |                 |
| 186 | 000124 | STATES | GETUIC          |
| 187 | 000124 | TRANS  | 'L, GTUIC       |
| 188 | 000124 | TRANS  | \$LAMDA, \$EXIT |
| 189 |        |        |                 |
| 190 | 000124 | STATES | GTUIC           |
| 191 | 000124 | TRANS  | 'J, \$EXIT      |
| 192 | 000124 | TRANS  | \$EOS, \$EXIT   |
| 193 | 000124 | TRANS  | \$ANY, GTUIC    |
| 194 |        |        |                 |
| 195 | 000124 | STATES |                 |

```

55 ;****
56 ; LOCAL MACROS
57 ;****
58 .MACRO PRINT TEXT
59 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
60 CALL $PRINT ; PRINT MESSAGE
61 .ENDM PRINT
62
63 ;****
64 ; MACRO CALLS
65 ;****
66 .MCALL ISTAT$,STATE$,TRAN$
67
68 ;****
69 ; LOCAL DATA
70 ;****
71 TEMP: .BLKW 1 ; ADDRESS OF FEATURES TEMPLATE
72 ;
73 ; DEFINE FEATURES TEMPLATE
74 ;
75 .ASECT
76 .=0
77 .BLKW 1 ; C.LNK
78 .BLKW 1 ; C.S'S
79 .BLKW 1 ; 2 CHARACTER DEVICE NAME
80 C.DEV: .BLKW 1 ; INITIAL FEATURES
81 C.WD1: .BLKW 1 ; STATES SETTABLE
82 C.WD2: .BLKW 1 ; CURRENT STATES
83 C.WD3: .BLKW 1
84 C.LEN=.
85 .PSECT
86
87 000000
88 000000
89 000002
90 000004
91 000006
92 000010
93 000012
94 000014

```

.TITLE TMPLLC - CFE PARSE LLC PROCESS DEFINITIONS IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE LLC DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RX V1.0

Symbol table

|         |         |     |          |          |          |          |         |          |          |          |          |          |     |
|---------|---------|-----|----------|----------|----------|----------|---------|----------|----------|----------|----------|----------|-----|
| BITS    | 000012R | 002 | FLAG     | 000000R  | NT\$RET= | 000032   | ZF.COU= | *****    | GX       | \$LAMDA= | 000000   |          |     |
| BITO    | 000024R | 002 | LLCKTB   | 000000RG | 003      | NT\$ROU= | 000024  | ZF.INI=  | *****    | GX       | \$MXNLT= | *****    | GX  |
| CL.LEN= | 000020  |     | LLCSTB   | 000000RG | 002      | NT\$RTR= | 000030  | ZF.LLC=  | *****    | GX       | \$NLN =  | *****    | GX  |
| CS.LLC= | *****   | GX  | NLT      | 000162R  | 002      | NT\$TSP= | 000026  | ZF.MFL=  | *****    | GX       | \$NUMB=  | 000002   |     |
| CTM     | 000216R | 002 | NM\$ARA= | 176000   |          | NM\$ACK= | 000011  | ZF.MTM=  | *****    | GX       | \$RAD50= | 000016   |     |
| C.CTM   | 000016  |     | NM\$NOD= | 001777   |          | NM\$HDR= | 000007  | ZF.SLI=  | *****    | GX       | \$STRNG= | 000004   |     |
| C.DEV   | 000012  |     | NT\$AKD= | 000020   |          | NM\$OVR= | 000022  | ZF.TIM=  | *****    | GX       | \$SUBXP= | 000010   |     |
| C.EXT   | 000015  |     | NT\$AKI= | 000022   |          | PRIOR    | 000130R | ZF.X3P=  | *****    | GX       | \$TALOC= | *****    | GX  |
| C.FLG   | 000006  |     | NT\$CC = | 000016   |          | SETCTM   | 000406R | \$ALPHA= | 000022   |          | \$TMLLC  | 000004RG |     |
| C.LIN   | 000011  |     | NT\$CON= | 000000   |          | SETDEV   | 000270R | \$ANY =  | 000020   |          | \$SFLG=  | 177777   |     |
| C.NAM   | 000004  |     | NT\$CTL= | 000000   |          | SETEXT   | 000356R | \$BLANK= | 000006   |          | \$SKEY=  | 000010   |     |
| C.NCTD= | 000013  |     | NT\$DAT= | 000002   |          | SETFLG   | 000134R | \$CAT5 = | *****    | GX       | \$SSTA=  | 000000   |     |
| C.NLT   | 000014  |     | NT\$DC = | 000012   |          | SETLIN   | 000222R | \$DIGIT= | 000024   |          | \$STMP=  | 000040R  | 004 |
| C.NLTD= | 000012  |     | NT\$DIS= | 000014   |          | SETNAM   | 000074R | \$DNUMB= | 000014   |          | .PNUMB=  | *****    | GX  |
| C.PRI   | 000010  |     | NT\$DLS= | 000006   |          | SETNLT   | 000326P | \$EOS =  | 000012   |          | .PNUMH=  | *****    | GX  |
| C.STS = | *****   | GX  | NT\$ILS= | 000010   |          | SETPRI   | 000164R | \$ERRUR= | *****    | GX       | .PSTCN=  | *****    | GX  |
| EXIT    | 000246R | 002 | NT\$IMS= | 000002   |          | TEMP     | 000002R | \$EXIT = | 000000   |          | .PSTPT=  | *****    | GX  |
| EXT     | 000200R | 002 | NT\$INT= | 000004   |          | ZF       | 000240R | 002      | \$FAIL = | 177777   | .TPARS=  | *****    | GX  |

. ABS. 000020 000 (RW,I,GBL,ABS,OVR)  
 000442 001 (RW,I,LCL,REL,CON)  
 \$STATE 000252 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000022 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000043 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10776 Words ( 43 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:45.24  
 SY:TMPLLC.V2,[132,134]TMPLLC/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPLLC

## SYMBOL CROSS REFERENCE

CREF 04.00

| SYMBOL    | VALUE       | REFERENCES         |
|-----------|-------------|--------------------|
| CS.LOG    | = ***** GX  | 7-131              |
| C.LEN     | = 000005    | #5-90 7-128        |
| C.STA     | = 000004    | #5-88 *8-164       |
| C.STS     | = ***** GX  | *7-131             |
| LA        | = 000074    | #5-78              |
| LOGKTB    | = 000000 RG | #6-99 7-132        |
| LOGSTB    | = 000000 RG | #6-99 7-133        |
| MAXSTA    | = 000007    | #5-77 8-162        |
| RA        | = 000076    | #5-79              |
| SETSTA    | = 000056 R  | #8-155             |
| TEMP      | = 000000 R  | #5-80 *7-130 8-155 |
| \$ALPHA   | = 000022    | #6-99              |
| \$ANY     | = 000020    | #6-99              |
| \$BLANK   | = 000006    | #6-99              |
| \$DIGIT   | = 000024    | #6-99              |
| \$DNUMB   | = 000014    | #6-99              |
| \$EOS     | = 000012    | #6-99              |
| \$ERROR   | = ***** GX  | *7-137             |
| \$EXIT    | = 000000    | #6-99              |
| \$FAIL    | = 177777    | #6-99              |
| \$GPRM    | = *****     | 6-99               |
| \$LAMDA   | = 000000    | #6-99              |
| \$NUMBER  | = 000002    | #6-99              |
| \$RAD50   | = 000016    | #6-99              |
| \$RONLY   | = *****     | 6-99 6-99          |
| \$STRNG   | = 000004    | #6-99              |
| \$SUBXP   | = 000010    | #6-99              |
| \$TALOC   | = ***** GX  | 7-129              |
| \$TMLOG   | = 000002 RG | #7-126             |
| \$\$\$FLG | = 177777    | #6-99              |
| \$\$\$KEY | = 177777    | #6-99              |
| .PNUMB    | = ***** GX  | 8-158              |
| .PNUMH    | = ***** GX  | 8-156              |
| .TFARS    | = ***** GX  | 7-135              |

325  
 326  
 327  
 328  
 329  
 330  
 331  
 332  
 333  
 334  
 335  
 336  
 337  
 338  
 339  
 340  
 341  
 342  
 343  
 344  
 345  
 346  
 347  
 348  
 349

000422  
 000422 016700 177:52  
 000426 016701 0000:00  
 000432 001406  
 000434 020127 001777  
 000440 101003  
 000442 050160 000054  
 000446 000402  
 000450 062716 000002  
 000454 000207  
 000001

```

;+
; *** - SETHOS - SET HOST NODE ADDRESS
; INPUT:
; .PNUMB = NODE ADDRESS
; .PNUMH = NODE ADDRESS (HIGH ORDER)
; OUTPUT:
; ADDRESS IS STORED IN C.HOS OF TEMPLATE
;-

SETHOS:
 MOV TEMP,R0 ; POINT R0 AT TEMPLATE
 MOV .PNUMB,R1 ; CHECK VALUE
 BEQ 5$, ; IF LE, ILLEGAL NODE NUMBER
 CMP R1,#1023. ; CHECK RANGE
 BHI 5$, ;
 BIS R1,C.HOS(R0) ; STORE THE HOST NODE ADDRESS
 BR 10$, ;
5$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
10$: RETURN

 .END

```

SETVFY - SETUP OBJECT VERIFICATION LEVEL

```

260 .SBTTL SETVFY - SETUP OBJECT VERIFICATION LEVEL
261 :+
262 :*** - SETVFY - SETUP OBJECT VERIFICATION LEVEL
263 :
264 :THIS ACTION ROUTINE SETS UP THE OBJECT VERIFICATION LEVEL.
265 :
266 :INPUT:
267 :.PNUMB = THE OBJECT VERIFICATION LEVEL
268 :.PNUMH = THE OBJECT VERIFICATION LEVEL (HIGH ORDER)
269 :TEMP = TEMPLATE ADDRESS
270 :
271 :OUTPUT:
272 :THE VERIFICATION LEVEL IS MOVED INTO C.FLG OF THE TEMPLATE
273 :C-BIT = SUCCESS/FAILURE
274 :
275 :-
276
277 000264 SETVFY:
278 000264 MOV TEMP,R0 ; POINT AT TEMPLATE
279 000270 MOV .PNUMB,C.FLG(R0) ; STORE THE VERIFICATION LEVEL
280 000276 TST .PNUMH ; IS THIS A WORD VALUE ?
281 000302 BNE 10$; IF NE, NO .. REJECT
282 000304 CMP #10,.PNUMB ; VERIFICATION LEVEL IS ONLY 3 BITS
283 000312 BHI 20$; IF HI, OKAY
284 000314 ADD #2,(SP) ; ELSE, REJECT TRANSITION
285 000320 10$: RETURN
285 000320 20$:

```



```

189 .SBTTL SETNMO - SET UP PARTITION NAME 0
190 :
191 : *** - SETNMO - SETUP PARTITION NAME 0
192 :
193 : THIS ACTION ROUTINE SETS UP THE PARTITION NAME 0.
194 :
195 : INPUT:
196 : .PSTPT = START ADDRESS OF THE PARTITION NAME 0 IN ASCII
197 : .PSTCN = NUMBER OF CHARACTERS IN THE PARTITION NAME 0
198 : TEMP = TEMPLATE ADDRESS
199 :
200 : OUTPUT:
201 : C.NAM0 = RAD50 PARTITION NAME 0 (2 WORDS)
202 :
203 : -
204
205 000104 SETNMO:
206 000104 016700 000000G MOV .PSTPT,R0 ; GET ADDRESS OF ASCII PARTITION NAME 0
207 000110 010701 MOV PC,R1 ; PERIODS ARE ACCEPTABLE
208 000112 004767 000000G CALL $CAT5 ; CONVERT PARTITION NAME 0 TO RAD50
209 000116 016702 177660 MOV TEMP,R2 ; GET START OF TEMPLATE
210 000122 010162 000004 MOV R1,C.NAM0(R2) ; STORE PARTITION NAME 0 IN TEMPLATE
211 000126 103407 BCS 5$; IF CS, LESS THAN 3 CHARS WERE RAD50
212 000130 010701 MOV PC,R1 ; ELSE, PERIODS ARE ACCEPTABLE
213 000132 004767 000000G CALL $CAT5 ; CONVERT THE REST OF THE PARTITION NAME
214 000136 016702 177640 MOV TEMP,R2 ; GET START OF TEMPLATE
215 000142 010162 000006 MOV R1,C.NAM0+2(R2) ; STORE THE SECOND WORD
216 000146 022767 000006 5$: CMP #6,.PSTCN ; IS THE PARTITION NAME 0 TOO BIG ?
217 000154 103002 BHIS 10$; IF HIS, NO
218 000156 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
219 000162 000207 10$: RETURN

```

```

126
127
128
129
130 000000
131 000000
132 000000
133 000002
134 000004
135 000010
136 000012
137 000014
138 000016
139 000020
140 000000
141
142
143
144 000004
145
146 000000 004 103
147 000042 004 103
148 000100 004 103
149 000127 004 103
150 000156 004 103
151
152

 .SBTTL LOCAL DATA
 ;
 ; DEFINE THE PSN BLOCK
 ;
 .ASECT
 .=0
 .BLKW 1 ; C.LNK
 .BLKW 1 ; C.STS
 C.NAM: .BLKW 2 ; RAD50 NETWORK NAME
 C.POR: .BLKW 1 ; NUMBER OF PORTS SUPPORTED (1-256)
 C.FLG: .BLKW 1 ; FLAGS
 C.TLO: .BLKW 1 ; TRANSPORT SUBADDRESS LOW
 C.THI: .BLKW 1 ; TRANSPORT SUBADDRESS HIGH
 C.LEN=. ; LENGTH OF THE BLOCK
 .PSECT
 ;
 ; ERROR TEXT STRINGS
 ;
 EX$SEV = 4
 .NLIST BEX
106 ERRPSN:.ASCIZ <EX$SEV>/CFE -- Duplicate PSN$DF in CETAB/
106 ERRNAM:.ASCIZ <EX$SEV>/CFE -- Error in network name/
106 ERRPOR:.ASCIZ <EX$SEV>/CFE -- Bad port value/
106 ERRFLG:.ASCIZ <EX$SEV>/CFE -- Error in flags/
106 ERRTRN:.ASCIZ <EX$SEV>/CFE -- Bad transport subaddress value/
 .EVEN
 .LIST BEX

```

114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126

```
MOV LEN,R1
; IF NB <OPT>
BIS OPT,@.FLAGS
; ENDC
JMP $GTR50
; ENDM

; MACRO CALLS
;
; .MCALL ISTAT$,STATE$,TRAN$,CALLR
```

TMPEVT

TMPLLO

TMPPSN

TMPRDT

.TITLE TMPRDT - PARSE REMOTE DTE IN CETAB  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE REMOTE DTE IN CETAB

#### DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/s V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSY V1.0

```

53 :****
54 : LOCAL MACROS
55 :****
56
57 .MACRO PRINT TEXT
58 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
59 CALL $PRINT ; PRINT MESSAGE
60 .ENDM
61
62 :****
63 : MACRO CALLS
64 :****
65
66 .MCALL ISTAT$,STAT$,TRANS$
67
68
69 :****
70 : LOCAL DATA
71 :****
72
73 LA = '<' ; LEFT ANGLE BRACKET
74 RA = '>' ; RIGHT ANGLE BRACKET
75 000000 000074 ; ADDRESS OF REMOTE TEMPLATE
76 000000 000076
77 : DEFINE REMOTE TEMPLATE
78 :
79 .ASECT
80 .=0
81 000000 000000
82 000002
83 000004
84 000010
85 000002 000012
86 000002
87 .PSECT

```

```

54 ;****
55 ; LOCAL MACROS
56 ;****
57
58 .MACRO PRINT TEXT
59 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
60 CALL $PRINT ; PRINT MESSAGE
61 .ENDM PRINT
62
63 .MACRO ERRORS$ TEXT
64 ;
65 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
66 ; MESSAGE STRING.
67 ;
68 .IF DIF <TEXT><R0>
69 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
70 .ENDC
71 CALL $CFERR ; PRINT ERROR MESSAGE
72 .ENDM ERRORS$
73
74 ;
75 ; GET NUMBER WITH RANGE CHECKING
76 ;
77 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
78 MOV OFFS,R2
79 MOV LO,R0
80 MOV HI,R1
81 .IF NB <OPT>
82 BIS OPT,@.FLAGS
83 .ENDC
84 JMP $GTNUM
85 .ENDM
86
87 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
88 MOV OFFS,R2
89 MOV LO,R0
90 MOV HI,R1
91 .IF NB <OPT>
92 BIS OPT,@.FLAGS
93 .ENDC
94 JMP $GTNMB
95 .ENDM
96
97 ;
98 ; GET RAD50 WITH FIELD SIZE CHECKING
99 ;
100 .MACRO $GTR50 OFFS,LEN,OPT
101 MOV OFFS,R2
102 MOV LEN,R1
103 .IF NB <OPT>
104 BIS OPT,@.FLAGS
105 .ENDC
106 JMP $GTR50
107 .ENDM
108
109 ;****
110 ; MACRO CALLS
111 ;****

```

```

47 ;****
48 ; MACRO CALLS
49 ;****
50
51 .MCALL ISTAT$,STATE$,TRAN$,SAVRG,RESRG
52
53
54 ;
55 ; DEFINE SER$DF TEMPLATE
56 ;
57 .ASECT
58 . = 0
59 .BLKW 1 ; C.LNK
60 .BLKW 1 ; C.FLG
61 SSFLG: .BLKW 1 ; option flags
62 S$ADD: .BLKW 1 ; Target node address (16 bit node address)
63 S$HST: .BLKW 1 ; Host node address (16 bit node address)
64 S$HAD: .BLKW 4 ; Hardware address (RAD50)
65 S$CIR: .BLKW 4 ; Service circuit (extended RAD50)
66 S$DEV: .BLKW 1 ; Service device (extended RAD50)
67 S$PSS: .BLKW 6 ; Service password (RAD50)
68 S$DPA: .BLKW 2 ; Dump address (octal binary)
69 S$DPC: .BLKW 1 ; Dump count (in k bytes)
70 S$LEN=.
71 .PSECT
72 ;
73 ; Flags definitions:
74 SF.ADD = 1 ; Target address specified
75 SF.HST = 2 ; Host address specified
76 SF.HAD = 4 ; Hardware address specified
77 SF.CIR = 10 ; Service circuit specified
78 SF.DEV = 20 ; Service device specified
79 SF.PSS = 40 ; Service password specified
80 SF.DPA = 100 ; Dump address specified
81 SF.DPC = 200 ; Dump count specified
82 SF.PH3 = 400 ; Service node type is Phase 3
83
84 ;
85 ; Local Data
86 ;
87 SERFLG: .BLKW 1
88 HEXSTR: .BLKB 16. ; HEX STRING BUFFER
89 ENDHEX

```



TMPSER      CREATED BY MACRO ON 29-JUN-85 AT 05:42      PAGE 2      B 5  
 SYMBOL CROSS REFERENCE      CREF 04.00

| SYMBOL  | VALUE      | REFERENCES                                                                                      |
|---------|------------|-------------------------------------------------------------------------------------------------|
| \$RONLY | = *****    | 5-96      5-96      5-96                                                                        |
| \$STRNG | = 000004   | #5-96                                                                                           |
| \$SUBXP | = 000010   | #5-96                                                                                           |
| \$TALOC | = ***** GX | 6-239                                                                                           |
| \$TMSER | 000022 RG  | #6-235                                                                                          |
| \$SFLG  | = 177777   | #5-96                                                                                           |
| \$SKEY  | = 177777   | #5-96                                                                                           |
| .CAT5E  | = ***** GX | 8-344      10-450                                                                               |
| .ERROR  | = ***** GX | *6-245                                                                                          |
| .FLAGS  | = ***** GX | *6-242                                                                                          |
| .PNUMB  | = ***** GX | 7-279      7-307      9-379      9-392                                                          |
| .PNUMH  | = ***** GX | 9-380                                                                                           |
| .FSTCN  | = ***** GX | 8-336      10-421                                                                               |
| .PSTPT  | = ***** GX | 8-341      9-364      10-427      10-447                                                        |
| .TEMP   | = ***** GX | *6-240      6-250      7-283      7-312      8-340      9-366      9-378      9-391      10-446 |
| .TPARS  | = ***** GX | 6-249                                                                                           |

TMPSLT - CFE PARSE LINE DEFINIT MACRO V05.03b Saturday 29-Jun-85 05:42 Page 10  
SETUNT - SET THE UNIT NUMBER

```
337 .SBITL SETUNT - SET THE UNIT NUMBER
338
339 :+
340 :*** - SETUNT - SET THE UNIT NUMBER
341 :
342 :INPUT:
343 :.PNUMB = UNIT NUMBER
344 :.PNUMH = UNIT NUMBER (HIGH ORDER)
345 :
346 :OUTPUT:
347 :THE UNIT NUMBER IS STORED IN S$LUNT OF THE TEMPLATE.
348 :
349 :-
350 SETUNT: $GTNMB #S$LUNT
351
```

TMPS

TMP

```

122 ;****
123 ; INITIALIZE TABLE GENERATION
124 ;****
125
126 000036 ISTAT$ STASTB,STAKTB
127
128 000036 STATES$
129 000036 TRANS$ $NUMBR,,SEIADD
130
131 000036 STATES$
132 000036 TRANS$ <','>
133
134 000036 STATES$
135 000036 TRANS$ $RAD50
136
137 000036 STATES$
138 000036 TRANS$ <','>,SFLAG
139 000036 TRANS$ $LAMDA,$EXIT
140
141 000036 STATES$ SFLAG
142 000036 TRANS$!SFS,STACST ; SKIP FLAG BITS
143 000036 TRANS$ <','>,STACST
144 000036 TRANS$ $LAMDA,$EXIT
145
146 000036 STATES$ STACST
147 000036 TRANS$ $NUMBR,,SETCST ; SET STATION COST
148
149 000036 STATES$
150 000036 TRANS$ <','>,APRNUM ; STATION ACTIVE POLLING RATIO
151 000036 TRANS$ $LAMDA,$EXIT
152
153 000036 STATES$ APRNUM
154 000036 TRANS$ $NUMBR,,SETAPR
155
156 000036 STATES$
157 000036 TRANS$ <','>,HTIMR ; HELLO TIMER
158 000036 TRANS$ $LAMDA,$EXIT
159
160 000036 STATES$ HTIMR
161 000036 TRANS$ $NUMBR,$EXIT,SETHM
162
163 ;
164 ; LOOK FOR BITS
165 ;
166 000036 STATES$ SFS
167 000036 TRANS$ "SF"
168
169 000036 STATES$
170 000036 TRANS$ '.,BITS
171
172 000036 STATES$ SKIP
173 000036 TRANS$ '!',SFS
174 000036 TRANS$ <','>,$EXIT
175
176 000036 STATES$ BITS
177 000036 TRANS$ "ENA",SKIP,,SF.ENA,FLAG
178 000036 TRANS$ "SVC",SKIP,,SF.SVC,FLAG

```

```

55 .SBTTL MACRO DEFINITIONS
56
57 ; LOCAL MACROS
58
59
60 .MACRO PRINT TEXT
61 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
62 CALL $PRINT ; PRINT MESSAGE
63 .ENDM PRINT
64
65 .MACRO ERROR$ TEXT
66
67 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
68 ; MESSAGE STRING.
69
70 .IF DIF <TEXT><R0>
71 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
72 .ENDC
73 CALL $CFERR ; PRINT ERROR MESSAGE
74 .ENDM ERROR$
75
76 .MACRO EPRINT TEXT
77
78 ; PRINT TEXT ON ERROR LUN
79
80 .IF DIF <TEXT><R0>
81 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
82 .ENDC
83 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
84 .ENDM EPRINT
85
86 ; GET NUMBER WITH RANGE CHECKING
87
88 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
89 MOV OFFS,R2
90 MOV LO,R0
91 MOV HI,R1
92 .IF NB <OPT>
93 BIS OPT,@.FLAGS
94 .ENDC
95 JMP $GTNUM
96 .ENDM
97
98 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
99 MOV OFFS,R2
100 MOV LO,R0
101 MOV HI,R1
102 .IF NB <OPT>
103 BIS OPT,@.FLAGS
104 .ENDC
105 JMP $GTNMB
106 .ENDM
107
108 ; GET RAD50 WITH FIELD SIZE CHECKING
109
110 .MACRO $GTR50 OFFS,LEN,OPT
111 MOV OFFS,R2

```

.TITLE TMPUNT - CFE PARSE UNIT DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE UNIT DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

TMPUNT      CREATED BY    MACRO    ON: 29-JUN-85 AT 05:45      PAGE 2      B 10  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-99      | #5-121 | #5-128 | #5-132 | #5-136 | #5-140 |        |        |        |        |
| ISTAT\$    | #4-68      | 5-99   |        |        |        |        |        |        |        |        |
| MTRAN\$    | #5-99      |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-59      |        |        |        |        |        |        |        |        |        |
| STAT\$     | #4-68      | 5-101  | #5-104 | #5-107 | #5-110 | #5-113 | #5-116 | #5-119 | #5-123 | #5-126 |
|            | #5-130     | #5-134 | #5-138 | #5-142 | #5-145 | #5-148 |        |        |        |        |
| TRANS      | #4-68      | #5-102 | #5-105 | #5-108 | #5-111 | #5-114 | #5-117 | #5-120 | #5-121 | #5-124 |
|            | #5-127     | #5-128 | #5-131 | #5-132 | #5-135 | #5-136 | #5-139 | #5-140 | #5-143 | #5-146 |

IMPX2P      CREATED BY MACRO ON 29-JUN-85 AT 05:45      PAGE 2      B 11  
SYMBOL CROSS REFERENCE      CREF 04.00  
SYMBOL VALUE      REFERENCES  
.TPARS = \*\*\*\*\* GX      7-258

TMPX29      CREATED BY    MACRO    ON 29-JUN-85 AT 05:46      PAGE 2      B 12  
MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|
| DBGTP\$ | #6-143 |        |        |        |        |        |
| EPRINT  | #4-78  |        |        |        |        |        |
| ERROR\$ | #4-67  | 7-195  |        |        |        |        |
| ISTAT\$ | #4-124 | 6-143  |        |        |        |        |
| MIRAN\$ | #6-143 |        |        |        |        |        |
| PRINT   | #4-62  |        |        |        |        |        |
| STAT\$  | #4-124 | 6-148  | #6-150 | #6-152 | #6-155 | #6-158 |
| TRANS   | #4-124 | #6-149 | #6-151 | #6-153 | #6-156 | #6-159 |
| \$GTNMB | #4-100 | 8-214  |        |        |        |        |
| \$GTNUM | #4-90  | 8-212  |        |        |        |        |
| \$GTR50 | #4-112 |        |        |        |        | #6-164 |

TMP  
SYM  
SYM  
\$GT  
\$LA  
\$NU  
\$RA  
\$RO  
\$ST  
\$SU  
\$TA  
\$TM  
\$\$  
\$\$  
\$.ER  
\$.TE  
\$.TP



TMPX3P      CREATED BY MACRO ON 29-JUN-85 AT 05:46      PAGE 2      B 13

SYMBOL CROSS REFERENCE

CREF 04.00

| SYMBOL  | VALUE      | REFERENCES       |
|---------|------------|------------------|
| \$GTNUM | = ***** GX | 8-324      8-326 |
| \$LAMDA | = 000000   | #6-172           |
| \$NUMBR | = 000002   | #6-172           |
| \$RAD50 | = 000016   | #6-172           |
| \$RONLY | = *****    | 6-172      6-172 |
| \$STRNG | = 000004   | #6-172           |
| \$SUBXP | = 000010   | #6-172           |
| \$TALOC | = ***** GX | 7-279            |
| \$TMX3P | 000610 RG  | #7-276           |
| \$SFLG  | = 177777   | #6-172           |
| \$SKEY  | = 177777   | #6-172           |
| .ERROR  | = ***** GX | 7-287            |
| .TEMP   | = ***** GX | *7-280           |
| .TPARS  | = ***** GX | 7-285            |

\*\*FILE\*\*ID\*\*UPDCHN

|            |    |          |          |          |    |            |      |      |      |
|------------|----|----------|----------|----------|----|------------|------|------|------|
| UU         | UU | PPPPPPP  | DDDDDDDD | CCCCCCCC | HH | HH         | NN   | NN   |      |
| UU         | UU | PPPPPPP  | DDDDDDDD | CCCCCCCC |    | HH         | NN   | NN   |      |
| UU         | UU | PP       | DD       | DD       | CC | h.         | NN   | NN   |      |
| UU         | UU | PP       | DD       | DD       | CC | HH         | NN   | NN   |      |
| UU         | UU | PP       | DD       | DD       | CC | HH         | NNNN | NN   |      |
| UU         | UU | PP       | DD       | DD       | CC | HH         | NNNN | NN   |      |
| UU         | UU | PPPPPPP  | DD       | DD       | CC | HHHHHHHHHH | NN   | NN   | NN   |
| UU         | UU | PPPPPPP  | DD       | DD       | CC | HHHHHHHHHH | NN   | NN   | NN   |
| UU         | UU | PP       | DD       | DD       | CC | HH         | NN   | NNNN |      |
| UU         | UU | PP       | DD       | DD       | CC | HH         | NN   | NNNN |      |
| UU         | UU | PP       | DD       | DD       | CC | HH         | NN   | NN   |      |
| UU         | UU | PP       | DD       | DD       | CC | HH         | NN   | NN   | .... |
| UU         | UU | PP       | DD       | DD       | CC | HH         | NN   | NN   | .... |
| UUUUUUUUUU | PP | DDDDDDDD | CCCCCCCC | HH       | HH | NN         | NN   | NN   | .... |
| UUUUUUUUUU | PP | DDDDDDDD | CCCCCCCC | HH       | HH | NN         | NN   | NN   | .... |

|            |          |           |
|------------|----------|-----------|
| LL         | SSSSSSSS | TTTTTTTTT |
| LL         | SSSSSSSS | TTTTTTTTT |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SSSSSS   | TT        |
| LL         | SSSSSS   | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LLLLLLLLLL | SSSSSSSS | TT        |
| LLLLLLLLLL | SSSSSSSS | TT        |

```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 ;****
65 ; MACRO CALLS
66 ;****
67
68 .MCALL ISTAT$,STAT$,TRAN$
69
70 ;****
71 ; LOCAL DATA
72 ;****
73
74 000000 TEMP: .BLKW 1 ; ADDRESS OF CONTROLLER TEMPLATE
75 ;
76 ; DEFINE CONTROLLER TEMPLATE
77 ;
78 000000 .ASECT
79 000000 .=0
80 000000 .BLKW 1 ; C.LNK
81 000002 .BLKW 1 ; C.STS
82 000004 C.CNT: .BLKW 1 ; CONTROLLER NUMBER
83 000006 C.VEC: .BLKW 1 ; VECTOR ADDRESS
84 000010 C.CSR: .BLKW 1 ; CSR ADDRESS
85 000012 C.PRI: .BLKW 1 ; CONTROLLER PRIORITY
86 000014 C.IRM: .BLKW 1 ; UNIBUS RUN MASK
87 000016 C.LEN=.
88 000002 .PSECT
89

```

.TITLE UPDCUG - UPDATE USER GROUP DEFINITION IN CETAB  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

UPDATE USER GROUP DEFINITION IN CETAB

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0

```

57 .SBTTL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61
62 .MACRO PRINT TEXT
63 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM PRINT
66
67 .MACRO ERRORS$ TEXT
68
69 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
70 ; MESSAGE STRING.
71
72 .IF DIF <TEXT><R0>
73 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
74 .ENDC
75 CALL $CFERR ; PRINT ERROR MESSAGE
76 .ENDM ERRORS$
77
78 .MACRO EPRINT TEXT
79
80 ; PRINT TEXT ON ERROR LUN
81
82 .IF DIF <TEXT><R0>
83 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
84 .ENDC
85 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
86 .ENDM EPRINT
87
88 ; GET NUMBER WITH RANGE CHECKING
89
90 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
91 MOV OFFS,R2
92 MOV LO,R0
93 MOV HI,R1
94 .IF NB <OPT>
95 BIS OPT,@.FLAGS
96 .ENDC
97 JMP $GTNUM
98 .ENDM
99
100 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
101 MOV OFFS,R2
102 MOV LO,R0
103 MOV HI,R1
104 .IF NB <OPT>
105 BIS OPT,@.FLAGS
106 .ENDC
107 JMP $GTNMB
108 .ENDM
109
110 ; GET RAD50 WITH FIELD SIZE CHECKING
111
112 .MACRO $GTR50 OFFS,LEN,OPT
113 MOV OFFS,R2

```

```

89 :****
90 : INITIALIZE TABLE GENERATION
91 :****
92
93
94 000002 ISTAT$ REMSTB,REMKTB
95
96
97 000002 STATES$
98 000002 TRANS$ LA
99
100 000002 STATES$
101 000002 TRANS$ $STRNG,,SETNAM
102
103 000002 STATES$
104 000002 TRANS$ RA
105
106 000002 STATES$
107 000002 TRANS$ <','>
108
109 000002 STATES$
110 000002 TRANS$ <','>,ADD2
111 000002 TRANS$ $NUMBR,,SETARE
112
113
114 000002 STATES$
115 000002 TRANS$ <','>
116
117 000002 STATES$ ADD2
118 000002 TRANS$ $NUMBR,,SETADD
119
120 : EXIT STATE
121 :
122 000002 STATES$ EXIT
123 000002 TRANS$ $EOS,$EXIT
124
125 000002 STATES$

```

```

111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140

```

```

.MCALL ISTAT$,STATE$,TRANS$,CALLR

;****
; LOCAL DATA
;****

LA = '<' ; LEFT ANGLE BRACKET
RA = '>' ; RIGHT ANGLE BRACKET
MAXRTM = 65535. ; ROUTING TIMER MAXIMUM VALUE
TEMP: .BLKW 1 ; ADDRESS OF REMOTE TEMPLATE

; DEFINE ROUTING TEMPLATE
;
.ASECT
.=0
.BLKW 1 ; C.LNK
.BLKW 1 ; C.STS
C.ADD: .BLKW 1 ; MAXIMUM NODE ADDRESS
C.CST: .BLKW 1 ; MAXIMUM COST ALLOWED
C.HOP: .BLKW 1 ; MAXIMUM NUMBER OF HOPS ALLOWED
C.RTM: .BLKW 1 ; ROUTING TIMER
C.NA: .BLKW 1 ; NUMBER OF AREAS
C.NBEA: .BLKW 1 ; NUMBER BRO ENDNODE ADJACENCIES
C.AHOP: .BLKW 1 ; AREA HOPS
C.ACST: .BLKW 1 ; AREA COST
C.L2TM: .BLKW 1 ; LEVEL 2 TIMER
C.LEN=.
.PSECT

```

```

000074
000076
177777
000000
000000
000000
000002
000004
000006
000010
000012
000014
000016
000020
000022
000024
000026
000002

```

```

91 ;****
92 ; INITIALIZE TABLE GENERATION
93 ;****
94
95 000022 ISTAT$ SERSTB,SERKTB
96
97 000022 STATES$
98 000022 TRANS$ $NUMBR, SER1, SETARE ; Remote area
99 000022 STATES$ SER1
100 000022 TRANS$ <','>
101
102 000022 STATES$
103 000022 TRANS$ $NUMBR,, SETADD, $F.ADD, SERFLG ; Remote address
104 000022 STATES$
105 000022 TRANS$ $EOS, EXIT
106 000022 TRANS$ <','>
107
108 000022 STATES$
109 000022 TRANS$ $NUMBR, SER5, HOSARE, $F.HST, SERFLG ; Optional Host area
110 000022 TRANS$ $LAMDA
111 000022 STATES$ SER5
112 000022 TRANS$ <','>
113
114 000022 STATES$
115 000022 TRANS$ $NUMBR, SER10, HOSADD, $F.HST, SERFLG ; Optional Host address
116 000022 TRANS$ $LAMDA
117 000022 STATES$ SER10
118 000022 TRANS$ $EOS, EXIT
119 000022 TRANS$ <','>
120
121 000022 STATES$
122 000022 TRANS$!CIRC, SER15 STCIR, $F.CIR, SERFLG ; Optional service circuit
123 000022 TRANS$ $LAMDA
124 000022 STATES$ SER15
125 000022 TRANS$ $EOS, EXIT
126 000022 TRANS$ <','>
127
128 000022 STATES$
129 000022 TRANS$ $STRNG, SER20, SERDV, $F.DEV, SERFLG ; Optional service device
130 000022 TRANS$ $LAMDA
131 000022 STATES$ SER20
132 000022 TRANS$ $EOS, EXIT
133 000022 TRANS$ <','>
134
135 000022 STATES$
136 000022 TRANS$ $STRNG, SER25, PAS, $F.PSS, SERFLG ; Optional password
137 000022 TRANS$ $LAMDA
138 000022 STATES$ SER25
139 000022 TRANS$ $EOS, EXIT
140 000022 TRANS$ <','>
141
142 000022 STATES$
143 000022 TRANS$ $NUMBR, SER30, DADDR, $F.DFA, SERFLG ; Optional dump address
144 000022 TRANS$ $LAMDA
145 000022 STATES$ SER30
146 000022 TRANS$ $EOS, EXIT
147

```



TMPSER CREATED BY MACRO ON 29-JUN-85 AT 05:42 PAGE 3 C 5

MACRO CROSS REFERENCE CREF 04.00

| MACRO NAME | REFERENCES                                                            |
|------------|-----------------------------------------------------------------------|
| DBGTP\$    | #5-96 #5-107 #5-111 #5-117 #5-120 #5-124 #5-127 #5-131 #5-134 #5-138  |
|            | #5-141 #5-145 #5-148 #5-152 #5-155 #5-159 #5-162 #5-166 #5-169 #5-181 |
|            | #5-191 #5-195 #5-212                                                  |
| STAT\$     | #4-51 5-96                                                            |
| TRAN\$     | #5-96                                                                 |
| RESRG      | #4-51 8-348 10-457                                                    |
| SAVRG      | #4-51 8-338 10-444                                                    |
| STAT\$     | #4-51 5-98 #5-100 #5-103 #5-105 #5-109 #5-112 #5-115 #5-118 #5-122    |
|            | #5-125 #5-129 #5-132 #5-136 #5-139 #5-143 #5-146 #5-150 #5-153 #5-157 |
|            | #5-160 #5-164 #5-167 #5-175 #5-177 #5-179 #5-183 #5-186 #5-189 #5-193 |
|            | #5-197 #5-199 #5-202 #5-204 #5-210 #5-217 #5-220                      |
| TRAN\$     | #4-51 #5-99 #5-101 #5-104 #5-106 #5-107 #5-110 #5-111 #5-113 #5-116   |
|            | #5-117 #5-119 #5-120 #5-123 #5-124 #5-126 #5-127 #5-130 #5-131 #5-133 |
|            | #5-134 #5-137 #5-138 #5-140 #5-141 #5-144 #5-145 #5-147 #5-148 #5-151 |
|            | #5-152 #5-154 #5-155 #5-158 #5-159 #5-161 #5-162 #5-165 #5-166 #5-168 |
|            | #5-169 #5-176 #5-178 #5-180 #5-181 #5-184 #5-187 #5-190 #5-191 #5-194 |
|            | #5-195 #5-198 #5-200 #5-203 #5-205 #5-211 #5-212 #5-218               |

TMPSLT - CFE PARSE LINE DEFINIT MACRO V05.03b Saturday 29-Jun-85 05:42<sup>6</sup> Page 11  
SETCTT - SET THE COUNTER TIMER

```
353 .SBTTL SETCTT - SET THE COUNTER TIMER
354 ;+
355 ;*** - SETCTT - SET THE COUNTER TIMER
356 ;
357 ; INPUT:
358 ; .PNUMB = COUNTER TIMER
359 ;
360 ; OUTPUT:
361 ; THE COUNTER TIMER IS STORED IN S$LCIT OF THE TEMPLATE.
362 ; -
363
364 000576 SETCTT: $GTNUM #S$LCIT,,,#SL$CTM
```

179 000036  
180 000036  
181  
182 000036

TRANS 'MFL',SKIP,,SF.MFL,FLAG  
TRANS 'SER',SKIP,,SF.SER,FLAG  
STATE\$

```
112 MOV LEN,R1
113 IF NB <OPT>
114 BIS OPT,@.FLAGS
115 .ENDC
116 JMP $GTR50
117 .ENDM
118
119 ;
120 ; MACRO CALLS
121 ;
122 .MCALL ISTAT$,STATE$,TRAN$,CALLR,SVCDF$
123
124 00000C SVCDF$; DEFINE SVC FLAGS
125
126
```

```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 ;****
65 ; MACRO CALLS
66 ;****
67
68 .MCALL ISTAT$,STAT$,TRANS$
69
70 ;****
71 ; LOCAL DATA
72 ;****
73
74 000000 TEMP: .BLKW 1 ; ADDRESS OF UNIT TEMPLATE
75 ;
76 ; DEFINE UNT TEMPLATE
77 ;
78 000000 .ASECT
79 000000 .=0
80 000000 .BLKW 1 ; C.LNK
81 000002 .BLKW 1 ; C.STS
82 000004 C.UNT: .BLKW 1 ; UNIT NUMBER
83 000006 C.CWO: .BLKW 1 ; CHARACTERISTICS WORD 0
84 000010 C.CWI: .BLKW 1 ; CHARACTERISTICS WORD 1
85 000012 C.CSR: .BLKW 1 ; SECONDARY CSR ADDRESS
86 000014 C.CST: .BLKW 1 ; LINE COST
87 000016 C.DPR: .BLKW 1 ; DEAD POLLING RATIO
88 000020 .BLKW 1 ; UNUSED
89 000022 .BLKW 1 ; UNUSED
90 000024 C.PCH: .BLKW 1 ; EMULATOR CHARACTERISTICS
91 000026 C.LEN=.
92 000002 .PSECT
93

```

\*\*\*FILE\*\*ID\*\*TMPX2P

C 10

```
TTTTTTTTTT MM MM PPPPPPP XX XX 222222 PPPPPPP
TTTTTTTTTT MM MM PPPPPPP XX XX 222222 PPPPPPP
TT MM MM PP PP XX XX 22 22 PP PP
TT MMMM MMMM PP PP XX XX 22 22 PP PP
TT MM MM MM PP PP XX XX 22 22 PP PP
TT MM MM MM PPPPPPP XX XX 22 22 PPPPPPP
TT MM MM PPPPPPP XX XX 22 22 PPPPPPP
TT MM MM PP PP XX XX 22 22 PP PP
TT MM MM PP XX XX 22 22 PP PP
TT MM MM PP XX XX 22 22 PP PP
TT MM MM PP XX XX 2222222222 PP
TT MM MM PP XX XX 2222222222 PP
```

....  
....  
....  
....

```
LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
```

TMPX2P      CREATED BY    MACRO    ON 29-JUN-85 AT 05:45      PAGE 3      C 11  
 MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| CALLR   | #4-124 |        |        |        |        |        |        |        |        |        |
| DBGTP\$ | #6-164 | #6-221 |        |        |        |        |        |        |        |        |
| EPRINT  | #4-78  |        |        |        |        |        |        |        |        |        |
| ERPOR\$ | #4-67  | 7-262  |        |        |        |        |        |        |        |        |
| ISTAT\$ | #4-124 | 6-164  |        |        |        |        |        |        |        |        |
| MTRAN\$ | #6-164 |        |        |        |        |        |        |        |        |        |
| PRINT   | #4-62  |        |        |        |        |        |        |        |        |        |
| STAT\$  | #4-124 | 6-168  | #6-170 | #6-172 | #6-175 | #6-177 | #6-179 | #6-182 | #6-184 | #6-186 |
|         | #6-189 | #6-191 | #6-193 | #6-196 | #6-198 | #6-200 | #6-203 | #6-205 | #6-207 | #6-213 |
|         | #6-215 | #6-217 | #6-219 | #6-222 | #6-227 |        |        |        |        |        |
| TRAN\$  | #4-124 | #6-169 | #6-171 | #6-173 | #6-176 | #6-178 | #6-180 | #6-183 | #6-185 | #6-187 |
|         | #6-190 | #6-192 | #6-194 | #6-197 | #6-199 | #6-201 | #6-204 | #6-206 | #6-208 | #6-214 |
|         | #6-216 | #6-218 | #6-220 | #6-221 | #6-223 |        |        |        |        |        |
| \$GTNMB | #4-100 |        |        |        |        |        |        |        |        |        |
| \$GTNUM | #4-90  | 8-286  | 8-288  | 8-290  | 8-292  | 8-294  | 8-312  | 8-314  |        |        |
| \$GTR50 | #4-112 | 8-310  |        |        |        |        |        |        |        |        |

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
 SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```



IMPX3P      CREATED BY    MACRO    ON 29-JUN-85 AT 05:46      PAGE 3      C 13  
 MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$ | #6-172 |        |        |        |        |        |        |        |        |        |
| EPRINT  | #4-78  |        |        |        |        |        |        |        |        |        |
| ERROR\$ | #4-67  | 7-289  |        |        |        |        |        |        |        |        |
| ISTAT\$ | #4-124 | 6-172  |        |        |        |        |        |        |        |        |
| MTRAN\$ | #6-172 |        |        |        |        |        |        |        |        |        |
| PRINT   | #4-62  |        |        |        |        |        |        |        |        |        |
| STATE\$ | #4-124 | 6-177  | #6-179 | #6-181 | #6-184 | #6-186 | #6-188 | #6-191 | #6-193 | #6-195 |
|         | #6-198 | #6-200 | #6-202 | #6-205 | #6-207 | #6-209 | #6-212 | #6-214 | #6-216 | #6-219 |
|         | #6-221 | #6-223 | #6-226 | #6-228 | #6-230 | #6-233 | #6-235 | #6-237 | #6-240 | #6-242 |
|         | #6-244 | #6-247 | #6-249 | #6-252 | #6-258 |        |        |        |        |        |
| TRANS   | #4-124 | #6-178 | #6-180 | #6-182 | #6-185 | #6-187 | #6-189 | #6-192 | #6-194 | #6-196 |
|         | #6-199 | #6-201 | #6-203 | #6-206 | #6-208 | #6-210 | #6-213 | #6-215 | #6-217 | #6-220 |
|         | #6-222 | #6-224 | #6-227 | #6-229 | #6-231 | #6-234 | #6-236 | #6-238 | #6-241 | #6-243 |
|         | #6-245 | #6-248 | #6-250 | #6-253 |        |        |        |        |        |        |
| \$GTNMB | #4-100 | 8-328  | 8-330  | 8-332  | 8-334  | 8-336  | 8-338  | 8-340  | 8-342  | 8-344  |
| \$GTNUM | #4-90  | 8-324  | 8-326  |        |        |        |        |        |        |        |
| \$GTR50 | #4-112 |        |        |        |        |        |        |        |        |        |

UPDCHN - UPDATE OUTGOING CHANNE MACRO V05.03b Saturday 29-Jun-85 05:47 <sup>D.14</sup>  
Table of contents

4- 57 MACRO CALLS AND LOCAL DATA  
4- 58 MACRO DEFINITIONS  
5- 123 LOCAL DATA  
6- 132 \$UPCHN - UPDATE CHANNEL DEFINITION

```

91 ;****
92 ; INITIALIZE TABLE GENERATION
93 ;****
94
95 000002 ISTAT$ CNTSTB,CNTKTB
96
97 000002 STATES$
98 000002 TRANS$ $NUMBR,,SETCNT
99
100 000002 STATES$
101 000002 TRANS$ <','>,$MOVPT
102
103 000002 STATES$
104 000002 TRANS$ $NUMBR,,SETVEC
105
106 000002 STATES$
107 000002 TRANS$ <','>,$MOVPT
108
109 000002 STATES$
110 000002 TRANS$ $NUMBR,,SETCSR
111
112 000002 STATES$
113 000002 TRANS$ <','>,$MOVPT
114
115 000002 STATES$
116 000002 TRANS$ $NUMBR,,SETPRI
117
118 000002 STATES$
119 000002 TRANS$ <','>,$MOVPT
120
121 000002 STATES$
122 000002 TRANS$ $NUMBR,RCDEV,SETURM
123 000002 TRANS$ $LAMDA,,SETURM
124
125 000002 STATES$ RCDEV
126 000002 TRANS$ <','>,$MOVPT
127
128 000002 STATES$
129 000002 TRANS$ $ALPHA,,$MOVPT
130 000002 STATES$
131 000002 TRANS$ $ALPHA,,$MOVPT
132
133 000002 STATES$
134 000002 TRANS$ $EOS,$EXIT,$CLNUP
135
136 000002 STATES$

```

```

57 .SBTTL MACRO CALLS AND LOCAL DATA
58 .SBTTL MACRO DEFINITIONS
59
60 ; LOCAL MACROS
61 ;
62
63 .MACRO PRINT TEXT
64 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
65 CALL $PRINT ; PRINT MESSAGE
66 .ENDM
67
68 .MACRO ERROR$ TEXT
69
70 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
71 ; MESSAGE STRING.
72 ;
73 .IF DIF <TEXT><R0>
74 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
75 .ENDC
76 CALL $CFERR ; PRINT ERROR MESSAGE
77 .ENDM
78
79 .MACRO EPRINT TEXT
80
81 ; PRINT TEXT ON ERROR LUN
82 ;
83 .IF DIF <TEXT><R0>
84 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
85 .ENDC
86 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
87 .ENDM
88 EPRINT
89
90 ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
91 ;
92 .MACRO $$$PTN ROU
93 .MACRO 'ROU OFS,OCTL,OPT,?L
94 .IF NB <OPT>
95 MOV OPT,R2
96 CALL $PTOPT
97 BCS L
98 .IFTF
99 MOV OFS,R2
100 .IF NB <OCTL>
101 MOV SP,R1
102 .IFF
103 CLR R1
104 .ENDC ; IF NB <OCTL>
105 CALL 'ROU
106 .IFT ; IF NB <OPT>
107
108 .ENDC ; IF NB <OPT>
109 .ENDM ; 'ROU
110 .ENDM ; $$$PTN
111
112 $$$PTN $PTNUM
113 $$$PTN $PTNMB
114
115 ;
116
117 000000
118 000000
119
120

```

```
114 MOV LEN,R1
115 .IF NB <OPT>
116 BIS OPT,a.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 ;
122 ; MACRO CALLS
123 ;
124 .MCALL ISTAT$,STAT$,TRANS$,CALLR
125
```

```

127 .SBTTL $TMREM - SETUP REMOTE TEMPLATE
128
129 *** - $TMREM - SETUP REMOTE TEMPLATE
130
131 INPUT:
132 R3-R5 - TPARS REGISTERS
133
134 OUTPUT:
135 C-BIT = SUCCESS/FAILURE
136 IF SUCCESS, THE REMOTE TEMPLATE IS STORED IN THE END OF TASK BUFFER.
137
138
139
140 $TMREM::
141 000002 010546 MOV R5, -(SP) ; SAVE R5
142 000004 012701 000012 MOV #C.LEN, R1 ; GET LENGTH OF ALLOCATION
143 000010 004767 000000G CALL $T1ALC ; TRY TO ALLOCATE CORE BLOCK
144 000014 010067 177760 MOV R0, TEMP ; STORE TEMPLATE ADDRESS
145 000020 016701 000000G MOV $NIADD, R1 ; GET EXEC ADDRESS
146 000024 042701 001777 BIC #1777, R1 ; ISOLATE AREA
147 000030 010160 000010 MOV R1, C.ADD(R0) ; INITIALIZE WITH DEFAULT AREA
148 000034 012760 000000G 000000G MOV #CS.REM, C.STS(R0) ; INDICATE THAT THIS IS A REM$DF TEMPLATE
149 000042 005001 CLR R1 ; IGNORE BLANKS
150 000044 012702 000000' MOV #REMKTB, R2 ; KEYWORD TABLE
151 000050 012705 000000' MOV #REMSTB, R5 ; STATE TABLE
152 000054 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
153 000060 103002 BCC 20$; NORMAL RETURN IF NO ERROR
154 000062 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
155 000066 016700 177706 20$: MOV TEMP, R0 ; RETRIEVE TEMPLATE ADDRESS
156 000072 004767 000016 CALL FINREM ; FIND A PLACE TO STORE REMOTE TEMPLATE
157 000076 004767 000000G CALL $QINSN ; INSERT INTO LIST
158 000102 062767 000000G 000000G ADD #R.LENX, $BYTMN ; UPDATE MINIMUM POOL BYTE EXTENSION
159 000110 012605 MOV (SP)+, R5 ; RESTORE R5
160 000112 000207 RETURN

```

```

142 ;****
143 ; INITIALIZE TABLE GENERATION
144 ;****
145
146 ISTAT$ ROUSTB,ROUKTB
147 000002
148
149 STATES$
150 000002 TRANS$ $NUMBR,,SETADD
151 000002
152
153 STATES$
154 000002 TRANS$ <','>
155
156 STATES$
157 000002 TRANS$ $NUMBR,,SETCST
158
159 STATES$
160 000002 TRANS$ <','>
161
162 STATES$
163 000002 TRANS$ $NUMBR,,SETHOP
164
165 STATES$
166 000002 TRANS$ $EOS,$EXIT
167 000002 TRANS$ <','>
168
169 STATES$
170 000002 TRANS$ $NUMBR,,SETRTM
171
172 STATES$
173 000002 TRANS$ $EOS,$EXIT
174 000002 TRANS$ <','>
175 000002 STATES$
176 000002 TRANS$ $NUMBR,,SETNA
177
178 STATES$
179 000002 TRANS$ <','>
180 000002 STATES$
181 000002 TRANS$ $NUMBR,,SENBEA
182
183 STATES$
184 000002 TRANS$ <','>
185 000002 STATES$
186 000002 TRANS$ $NUMBR,,SEAHOP
187
188 STATES$
189 000002 TRANS$ <','>
190 000002 STATES$
191 000002 TRANS$ $NUMBR,,SEACST
192
193 STATES$
194 000002 TRANS$ <','>
195 000002 STATES$
196 000002 TRANS$ $NUMBR,,SELV2T
197
198

```

```

148 000022 TRANS <','>
149
150 000022 STATES$; Optional dump count
151 000022 TRANS$ $NUMBR, SER35, DCT $F.DPC, SERFLG
152 000022 TRANS$ $LAMBDA
153 000022 STATES$ SER35
154 000022 TRANS$ $EOS, EXIT
155 000022 TRANS$ <','>
156
157 000022 STATES$; Optional hardware address
158 000022 TRANS$ $STRNG, SER40, STHAD, $F.HAD, SERFLG
159 000022 TRANS$ $LAMBDA
160 000022 STATES$ SER40
161 000022 TRANS$ $EOS, EXIT
162 000022 TRANS$ <','>
163
164 000022 STATES$ SER45 ; Optional Service node type
165 000022 TRANS$ %SF.PH3%,SER50, , $F.PH3, SERFLG
166 000022 TRANS$ %SF.PH4%
167 000022 STATES$ SER50
168 000022 TRANS$ <'>, SER45
169 000022 TRANS$ $LAMBDA, EXIT
170
171 ;
172 ; Subexpression to parse service circuit
173 ;
174
175 000022 STATES$ CIRC ; x
176 000022 TRANS$!ALPNUM
177 000022 STATES$
178 000022 TRANS$!ALPNUM ; xx
179 000022 STATES$
180 000022 TRANS$!ALPNUM,CIRC5 ; xxx
181 000022 TRANS$ $LAMBDA ; or xx
182
183 000022 STATES$ CIRC5 ; xxx-
184 000022 TRANS$ <'>->
185
186 000022 STATES$
187 000022 TRANS$ $DNUMB ; xxx-nn
188
189 000022 STATES$
190 000022 TRANS$!CIRC20,CIRC10 ; xxx-nn["m]
191 000022 TRANS$ $LAMBDA
192
193 000022 STATES$ CIRC10
194 000022 TRANS$!CIRC30,$EXIT ; xxx-nn[-mm][.tt]
195 000022 TRANS$ $LAMBDA,$EXIT
196
197 000022 STATES$ CIRC20
198 000022 TRANS$ <'>-> ; parse optional mux unit
199 000022 STATES$
200 000022 TRANS$ $DNUMB, $EXIT
201
202 000022 STATES$ CIRC30 ; parse optional trib number
203 000022 TRANS$ <'>
204 000022 STATES$

```



\*\*FILE\*\*ID\*\*TMPSLT

```

TTTTTTTTTT MM MM PPPPPPPP SSSSSSSS LL TTTTTTTTTT
TTTTTTTTTT MM MM PPPPPPPP SSSSSSSS LL TTTTTTTTTT
TT MMMM MMMM PP PP SS LL TT
TT MM/MM MM/MM PP PP SS LL TT
TT MM MM MM PP PP SS LL TT
TT MM MM MM PP PP SS LL TT
TT MM MM PPPPPPPP SSSSSS LL TT
TT MM MM PPPPPPPP SSSSSS LL TT
TT MM MM PP SS LL TT
TT MM MM PP SS LL TT
TT MM MM PP SS LL TT
TT MM MM PP SS LL TT
TT MM MM PP SSSSSSSS LLLLLLLLLL TT
TT MM MM PP SSSSSSSS LLLLLLLLLL TT

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL S TT
LLLLLLLLLLL SSSSSSSS TT
LLLLLLLLLLL SSSSSSSS TT

```

```

366 .SBTTL SETHTM - SET THE HELLO TIMER
367 .SBTTL SETRPR - SET ROUTING PRIORITY
368 .SBTTL SETBRA - SET BROADCAST ROUTER ADJACENCIES
369
370 :+
371 : *** - SETHTM - SET THE HELLO TIMER
372 : *** - SETRPR - SET ROUTING PRIORITY
373 : *** - SETBRA - SET BROADCAST ROUTER ADJACENCIES
374 :
375 : INPUT:
376 : .PNUMB = HELLO TIMER
377 :
378 : OUTPUT:
379 : THE HELLO TIMER IS STORED IN S$LHTM OF THE TEMPLATE.
380 :-
381 SETHTM: $GTNUM #S$LHTM,,,#SL$HTM
382
383 SETRPR: $GTNUM #S$LRPR,,,#SL$RPR
384
385 SETBRA: $GTNUM #S$LBRA,,,#SL$BRA
386

```

```

184 .SBTTL $TMSTA - SETUP STATION TEMPLATE
185 :+
186 *** - $TMSTA - SETUP STATION TEMPLATE
187 :
188 INPUT:
189 R3-R5 - TPARS REGISTERS
190 :
191 OUTPUT:
192 C-BIT = SUCCESS/FAILURE
193 IF SUCCESS, THE STATION TEMPLATE IS STORED IN THE END OF TASK BUFFER.
194 :
195 :-
196
197 000036 $TMSTA::
198 000036 010546 MOV R5,-(SP) ; SAVE R5
199 000040 005067 177736 CLR FLAG ; INITIALIZE STATION FLAGS
200 000044 012701 000016 MOV #C.LEN,R1 ; GET LENGTH OF ALLOCATION
201 000050 004767 000000G CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
202 000054 010067 177720 MOV R0,TEMP ; STORE TEMPLATE ADDRESS
203 000060 012760 000000G 000000G MOV #CS.STA,C.STS(R0) ; INDICATE THAT THIS IS A STA$DF TEMPLATE
204 000066 105060 000007 CLRB C.CST(R0) ; CLEAR COST FIELD
205 000072 105060 000010 CLRB C.APR(R0) ; AND APR
206 000076 005001 CLR R1 ; IGNORE BLANKS
207 000100 012702 000000' MOV #S$AKTB,R2 ; KEYWORD TABLE
208 000104 012705 000000' MOV #STASTB,R5 ; STATE TABLE
209 000110 004767 000000G CALL .TPARS ; PARSE THE REST OF THE LINE
210 000114 103002 BCC 20$; NORMAL RETURN IF NO ERROR
211 000116 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
212 000122 012605 MOV (SP)+,R5 ; RESTORE R5
213 000124 000207 RETURN

```

```

128 .SBTTL LOCAL DATA
129 .NLIST BEX
130 ;
131 ; SYNTAX ERROR DETAIL STRINGS
132 ;
133 .NLIST BEX
134 EX$SEV = 4
135 000000 004 103 106 ERRNAM: .ASCIZ <EX$SEV>/CFE -- Error in SVC name/
136 000032 004 103 106 ERRPSZ: .ASCIZ <EX$SEV>/CFE -- Error in packet size/
137 000067 004 103 106 ERRWSZ: .ASCIZ <EX$SEV>/CFE -- Error in window size/
138 000124 004 103 106 ERRFLG: .ASCIZ <EX$SEV>/CFE -- Error in flags/
139 000153 004 103 106 ERRRCL: .ASCIZ <EX$SEV>/CFE -- Error in maximum recalls/
140 000214 004 103 106 ERRTIM: .ASCIZ <EX$SEV>/CFE -- Error in recall timer/
141 000252 004 103 106 ERDTE: .ASCIZ <EX$SEV>/CFE -- Error in DTE address/
142 000307 004 103 106 ERROWN: .ASCIZ <EX$SEV>/CFE -- Error in owner/
143 .EVEN
144 .LIST BEX
145

```

```

95 ;****
96 ; INITIALIZE TABLE GENERATION
97 ;****
98
99 000002 ISTAT$ UNTSTB,UNTKTB
100
101 000002 STATES$
102 000002 TRANS$ $NUMBR,,SETUNT
103
104 000002 STATES$
105 000002 TRANS$ <','>
106
107 000002 STATES$
108 000002 TRANS$ $NUMBR,,SETCWO
109
110 000002 STATES$
111 000002 TRANS$ <','>
112
113 000002 STATES$
114 000002 TRANS$ $NUMBR,,SETCW1
115
116 000002 STATES$
117 000002 TRANS$ <','>,SECCSR
118
119 000002 STATES$ SECCSR
120 000002 TRANS$ $NUMBR,LINCST,SETCSR
121 000002 TRANS$ $LAMDA
122
123 000002 STATES$ LINCST
124 000002 TRANS$ <','>
125
126 000002 STATES$; LINE COST
127 000002 TRANS$ $NUMBR,LINDPR,SETCST
128 000002 TRANS$ $LAMDA
129
130 000002 STATES$ LINDPR
131 000002 TRANS$ $EOS,$EXIT
132 000002 TRANS$ <','>
133
134 000002 STATES$
135 000002 TRANS$ $NUMBR,PCHA,SETDPR ; DEAD POLLING RATIO
136 000002 TRANS$ $LAMDA
137
138 000002 STATES$ PCHA
139 000002 TRANS$ $EOS,$EXIT
140 000002 TRANS$ <','>
141
142 000002 STATES$
143 000002 TRANS$ $NUMBR,,SETPCH ; EMULATOR CHARACTERISTICS
144
145 000002 STATES$
146 000002 TRANS$ $EOS,$EXIT
147
148 000002 STATES$

```

|    |     |                                           |
|----|-----|-------------------------------------------|
| 4- | 57  | MACRO DEFINITIONS                         |
| 5- | 126 | LOCAL DATA                                |
| 6- | 159 | TPARS STATE TABLES                        |
| 7- | 230 | \$TMX2P - BUILD X2P TEMPLATE              |
| 8- | 269 | SETWSZ - SET WINDOW BLOCK SIZE            |
| 8- | 270 | SETBSZ - SET BLOCK SIZE                   |
| 8- | 271 | SETRTC - SET RETRANSMIT COUNT             |
| 8- | 272 | SETRTT - SET RETRANSMIT TIMER             |
| 8- | 273 | SETHBT - SET HOLDBACK TIMER               |
| 8- | 297 | SETDEV - SET DEVICE NAME IN LINE ID       |
| 8- | 298 | SETCON - SET CONTROLLER NUMBER IN LINE ID |
| 8- | 299 | SETUNI - SET UNIT NUMBER IN LINE ID       |

```

TTTTTTTTTT MM MM PPPPPPP XX XX 222222 999999
TTTTTTTTTT MM MM PPPPPPP XX XX 222222 999999
 TT MMMM MMMM PP XX XX 22 22 99 99
 TT MMMM MMMM PP PP XX 22 22 99 99
 TT MM MM PP PP XX XX 22 22 99 99
 TT MM MM PP PP XX XX 22 22 99 99
 TT MM MM PPPPPPP XX XX 22 22 99999999
 TT MM MM PPPPPPP XX XX 22 22 99999999
 TT MM MM PP XX XX 22 22 99 99
 TT MM MM PP XX XX 22 22 99 99
 TT MM MM PP XX XX 22 22 99 99
 TT MM MM PP XX XX 22 22 99 99
 TT MM MM PP XX XX 222222222 999999
 TT MM MM PP XX XX 222222222 999999
 TT MM MM PP XX XX 222222222 999999

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
 SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

|    |     |                                    |
|----|-----|------------------------------------|
| 4- | 57  | MACRO DEFINITIONS                  |
| 5- | 126 | LOCAL DATA                         |
| 6- | 167 | TPARS STATE TABLES                 |
| 7- | 261 | \$TMX3P - BUILD X3P TEMPLATE       |
| 8- | 295 | SETDFP - SET DEFAULT BLOCK SIZE    |
| 8- | 296 | SETMXP - SET MAXIMUM BLOCK SIZE    |
| 8- | 297 | SETDW - SET DEFAULT WINDOW SIZE    |
| 8- | 298 | SETMW - SET MAXIMUM WINDOW SIZE    |
| 8- | 299 | SETCT - SET CALL TIMER VALUE       |
| 8- | 300 | SETKT - SET CLEAR TIMER VALUE      |
| 8- | 301 | SETRT - SET RESET TIMER VALUE      |
| 8- | 302 | SETST - SET RESTART TIMER VALUE    |
| 8- | 303 | SETKM - SET MAXIMUM CLEAR VALUE    |
| 8- | 304 | SETRM - SET MAXIMUM RESETS VALUE   |
| 8- | 305 | SEISM - SET MAXIMUM RESTARTS VALUE |



\*\*FILE\*\*ID\*\*UPDBUF

```

UU UU PPPPPPP DDDDDDD BBBB8888 UU UU FFFFFFFF
UU UU PPPPPPP DDDDDDD BBBB8888 UU UU FFFFFFFF
UU UU PP PP DD DD BB BB UU UU FF
UU UU PP PP DD DD BB BB UU UU FF
UU UU PP PP DD DD BB BB UU UU FF
UU UU PP PP DD DD BB BB UU UU FF
UU UU PPPPPPP DD DD BBBB8888 UU UU FFFFFFFF
UU UU PPPPPPP DD DD BBBB8888 UU UU FFFFFFFF
UU UU PP PP DD DD BB BB UU UU FF
UU UU PP PP DD DD BB BB UU UU FF
UU UU PP PP DD DD BB BB UU UU FF
UU UU PP PP DD DD BB BB UU UU FF
UUUUUUUU PP DD DD BBBB8888 UUUUUUUU FF
UUUUUUUU PP DD DD BBBB8888 UUUUUUUU FF

```

```

....
....
....

```

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSS TT
LL SSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLL SSSSSSS TT
LLLLLLLL SSSSSSS TT

```

.TITLE UPDCHN - UPDATE OUTGOING CHANNEL DEFINITION IN CETAB  
.IDENT /V05.00/  
.ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

UPDATE OUTGOING CHANNEL DEFINITION IN CETAB

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-AUG-81  
DECNET-11M/S V3.1  
DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

```

138 .SBTTL $UPCNT - UPDATE CONTROLLER DEFINITION
139
140 *** - $UPCNT - UPDATE CONTROLLER DEFINITION
141
142 INPUT:
143 R3-R5 - TPARS REGISTERS
144
145 OUTPUT:
146 C-BIT = SUCCESS/FAILURE
147
148 -
149
150 $UPCNT::
151 000002 012700 000000G MOV $TMLST,R0 ; POINT AT START OF TEMPLATES
152 000006 010001 10$: MOV R0,R1 ; SAVE CURRENT TEMPLATE ADDRESS
153 000010 011000 MOV (R0),R0 ; GET NEXT TEMPLATE
154 000012 001436 BEQ 101$; IF EQ, END OF TEMPLATES
155 000014 122760 000000G 000000G CMPB #CS.CNT,C.STS(R0) ; IS THIS A CONTROLLER TEMPLATE ?
156 000022 001371 BNE 10$; IF NE, NO .. KEEP LOOKING
157
158 ; ASSUME THIS IS THE RIGHT CONTROLLER TEMPLATE
159
160 000024 011011 MOV (R0),(R1) ; REMOVE TEMPLATE FROM LIST
161 000026 010067 177746 MOV R0,TEMP ; SAVE TEMPLATE ADDRESS
162 000032 032760 000000G 000000G BIT #CS.MOD,C.STS(R0) ; WAS THIS TEMPLATE MODIFIED ?
163 000040 001004 BNE 15$; IF NE, YES .. UPDATE THE RECORD
164 000042 012767 000000G 000000G MOV #SBUF1,$OFDB+F.NRBD+2 ; ELSE, OUTPUT THE INPUT BUFFER
165 000050 000416 BR 30$; AND LEAVE
166 000052 004767 000000G 15$: CALL $MOVST ; MOVE FIRST FIELD INTO OUTPUT BUFFER
167 000056 010546 MOV R5,-(SP) ; SAVE R5
168 000060 005001 CLR R1 ; IGNORE BLANKS
169 000062 012702 000000' MOV #CNTKTBL,R2 ; KEYWORD TABLE
170 000066 012705 000000' MOV #CNTSTB,R5 ; STATE TABLE
171 000072 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
172 000076 103002 BCC 20$; NORMAL RETURN IF NO ERROR
173 000100 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
174 000104 012605 20$: MOV (SP)+,R5 ; RESTORE R5
175 000106 000207 30$: RETURN
176
177 ; ERROR CONDITIONS
178
179 000110 000167 000000G 101$: JMP .CONER ; CONSISTENCY ERROR

```

```

114 ; PUT RAD50 TEMPLATE FIELD TO BUFFER
115 ;
116 .MACRO $PTR50 OFS,WC,OPT,?L
117 .IF NB <OPT>
118 MOV OPT,R2
119 CALL $PTOPT
120 BCS L
121 .IFTF
122 MOV OFS,R2
123 MOV WC,R1
124 CALL $PTR50
125 .IFT
126 L:
127 .ENDC
128 .ENDM
129

```

TMPRDT - PARSE REMOTE DTE IN CE MACRO V05.03b Saturday 29-Jun-85 05:40 Page 5

LOCAL DATA

```
127 .SBTTL LOCAL DATA
128 .NLIST BEX
129
130 :: SYNTAX ERROR DETAIL STRINGS
131 ::
132 EX$SEV = 4
133 000000 004 103 106 ERRNAM: .ASCIZ <EX$SEV>/CFE -- Illegal DTE name/
134 000031 004 103 106 ERRADR: .ASCIZ <EX$SEV>/CFE -- Invalid DTE address/
135 .EVEN
136 .LIST BEX
```

```

162 .SBTTL FINREM - FIND NEXT REMOTE IN SORTED LIST
163 ;+
164 *** - FINREM - FIND NEXT REMOTE IN SORTED LIST
165 ;
166 INPUTS:
167 RO = TEMPLATE TO BE INSERTED
168 $LSNAD = LAST NODE ADDRESS IN LIST
169 $LSTMP = ADDRESS OF LAST TEMPLATE IN LIST
170 ;
171 OUTPUTS:
172 R1 = ADDRESS OF TEMPLATE TO PRECEDE CURRENT TEMPLATE
173 $LSNAD,$LSTMP UPDATED IF NODE INSERTED AT END OF LIST
174 ;
175 -
176
177 000114 FINREM: MOV $LSTMP,R1 ; ASSUME TEMPLATE GOES AT END OF LIST
178 000114 016701 000000G BEQ 5$; BR IF FIRST TIME THROUGH LIST
179 000120 001404 CMP C.ADD(RO),$LSNAD ; DOES THE TEMPLATE GO HERE?
180 000122 026067 000010 000000G BHI 20$; BR IF YES
181 000130 101016 MOV #STMLST,R2 ; GET START ADDRESS OF TEMPLATES
182 000132 012702 000000G 5$: MOV R2,R1 ; SAVE ADDRESS OF PREVIOUS
183 000136 010201 10$: MOV (R2),R2 ; GET NEXT TEMPLATE IN LIST
184 000140 011202 BEQ 20$; IF EQ, END OF LIST
185 000142 001411 CMPB #CS.REM,C.STS(R2) ; IS THIS A REMOTE TEMPLATE ?
186 000144 122762 000000G 000000G BNE 10$; IF NE, NO .. KEEP LOOKING
187 000152 001371 CMP C.ADD(RO),C.ADD(R2) ; ELSE, DOES TEMPLATE GO HERE ?
188 000154 026062 000010 000010 BHIS 10$; IF HIS, NO .. KEEP LOOKING
189 000162 103365 BR 30$; ELSE FOUND PLACE TO INSERT
190 000164 000405 MOV RO,$LSTMP ; SAVE ADDRESS OF LAST TEMPLATE
191 000166 010067 000000G 20$: MOV C.ADD(RO),$LSNAD ; SAVE LAST NODE ADDRESS
192 000172 016067 000010 000000G MOV C.ADD(RO),$LSNAD
193 000200 000207 30$: RETURN ; ELSE, OKAY

```

```
199
200 : EXIT STATE
201 :
202 : STATES$ EXIT
203 : TRANS$ $EOS,$EXIT
204 : TRANS$ $LAMDA,$EXIT
205
206 STATES$
```

```

205 000022 TRANS $DNUMB, $EXIT
206
207 ; Subexpression to match an alphanumeric character
208 ;
209 STATE$ ALPNUM
210 000022 TRANS $ALPHA, $EXIT
211 000022 TRANS $DIGIT, $EXIT
212 000022
213
214 ; EXIT STATE
215 ;
216 STATE$ EXIT
217 000022 TRANS $EOS, $EXIT
218 000022
219 STATE$
220 000022

```



TMPSLT - CFE PARSE LINE DEFINIT MACRO V05.03b Saturday 29-Jun-85 <sup>F 5</sup> 05:42  
Table of contents

|     |     |                                           |
|-----|-----|-------------------------------------------|
| 6-  | 250 | \$TMSLT - SETUP SLT TEMPLATE              |
| 7-  | 286 | SETNAM - SET UP PROCESS NAME              |
| 8-  | 303 | SETFLG - SETUP THE FLAGS WORD             |
| 9-  | 321 | SETCNT - SET THE CONTROLLER NUMBER        |
| 10- | 337 | SETUNT - SET THE UNIT NUMBER              |
| 11- | 353 | SETCTT - SET THE COUNTER TIMER            |
| 12- | 366 | SETHTM - SET THE HELLO TIMER              |
| 12- | 367 | SETRPR - SET ROUTING PRIORITY             |
| 12- | 368 | SETBRA - SET BROADCAST ROUTER ADJACENCIES |
| 13- | 388 | SETMST - SET MASTER TYPE                  |
| 13- | 389 | SETSLV - SET SLAVE TYPE                   |

```

388 .SBTTL SETMST - SET MASTER TYPE
389 .SBTTL SETSLV - SET SLAVE TYPE
390 :+
391 :*- SETMST - SET MASTER TYPE
392 :*- SETSLV - SET SLAVE TYPE
393 :
394 : INPUT NONE.
395 :
396 : OUTPUT THE LINE TYPE IS SET IN THE TEMPLATE.
397 :-
398 .ENABL LSB
399
400 000726 012701 000001 SETMST: MOV #1,R1 ; TYPE = MASTER
401 000732 000401 BR 10$; JOIN COMMON CODE
402
403 000734 005001 SETSLV: CLR R1 ; TYPE = SLAVE
404 000736 016700 10$: MOV .TEMP,R0 ; POINT TO TEMPLATE
405 000742 052760 000000G 000000G BIS #SL$TYP,C.STS(R0) ; REMEMBER OPTION
406 000750 110160 000000G MOVB R1,LTYP(R0) ; SET LINE TYPE
407 000754 000207 RETURN
408
409 000001 .END

```

```

215 .SBTTL SETADD - SETUP STATION ADDRESS
216 ;+
217 *** - SETADD - SETUP STATION ADDRESS
218 ;
219 THIS ACTION ROUTINE SETS UP THE STATION ADDRESS
220 ;
221 INPUT:
222 .PNUMB = STATION ADDRESS
223 .PNUMH = (HIGH ORDER)
224 $STAN = STATION NUMBER
225 ;
226 OUTPUT:
227 C.ADD = STATION ADDRESS
228 C.STA = STATION NUMBER
229 ;
230 -
231
232 SETADD:
233 000126 016700 177646 MOV TEMP,RO ; GET TEMPLATE ADDRESS
234 000132 026727 000000G 000000G CMP $STAN,$$MXSTA ; ARE THERE TOO MANY STATIONS ?
235 000140 103405 BLO 5$; IF LO, NO
236 000142 ERROR$ #ERRSTA ; ELSE, PRINT ERROR MESSAGE
237 000152 000416 BR 10$; AND REJECT
238 000154 116760 000000G 000004 5$: MOV $STAN,C.STA(RO) ; STORE THE NUMBER OF THIS STATION
239 000162 005267 000000G INC $STAN ; ONE MORE STATION
240 000166 116760 000000G 000005 MOV .PNUMB,C.ADD(RO) ; STORE THE STATION ADDRESS IN TEMPLATE
241 000174 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
242 000200 001003 BNE 10$; IF NE, NO - REJECT
243 000202 105767 000001G TST .PNUMB+1 ; IS THE STATION ADDRESS IN RANGE ?
244 000206 001402 BEQ 20$; IF EQ, YES - OKAY
245 000210 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
246 000214 000207 20$: RETURN

```

```

147 .SBTTL TPARS STATE TABLES
148 ;
149 ; INITIALIZE TABLES
150 ;
151 ;
152 000336 ISTAT$ SVCSTB, SVCKTB
153 ;
154 ; PROCESS "SVC$DF"
155 ;
156 000336 STATES$: SVC NAME DLM-XX.YY
157 000336 TRANS$ $LAMDA,, $CLERR,ERRNAM,,.ERROR
158 000336 STATES$
159 000336 TRANS$ 'DLM'
160 000336 STATES$
161 000336 TRANS$ '<'-'>'
162 000336 STATES$
163 000336 TRANS$ $DNUMB,,SETNM1
164 000336 STATES$
165 000336 TRANS$ '<'.'>'
166 000336 STATES$
167 000336 TRANS$ $DNUMB,,SETNM2
168 000336 STATES$
169 000336 TRANS$ '<','>,, $CLERR,ERRPSZ,,.ERROR
170
171 ;
172 000336 STATES$: OPTIONAL PACKET SIZE
173 000336 TRANS$ $NUMBR,SVC1,SETPSZ
174 000336 TRANS$ $LAMDA
175 000336 STATES$ SVC1
176 000336 TRANS$ '<','>,, $CLERR,ERRWSZ,,.ERROR
177
178 ;
179 000336 STATES$: OPTIONAL WINDOW SIZE
180 000336 TRANS$ $NUMBR,SVC2,SETWSZ
181 000336 TRANS$ $LAMDA
182 000336 STATES$ SVC2
183 000336 TRANS$ '<','>,, $CLERR,ERRFLG,,.ERROR
184
185 ;
186 000336 STATES$: FLAGS
187 000336 TRANS$ $NUMBR,,SETFLG
188 000336 STATES$
189 000336 TRANS$ '<','>,, $CLERR,ERRRCL,,.ERROR
190
191 ;
192 000336 STATES$: MAXIMUM RECALLS
193 000336 TRANS$ $NUMBR,,SETRCL
194 000336 STATES$
195 000336 TRANS$ '<','>,, $CLERR,ERRTIM,,.ERROR
196
197 ;
198 000336 STATES$: RECALL TIMER
199 000336 TRANS$ $NUMBR,,SETTIM
200 000336 STATES$
201 000336 TRANS$ '<','>,, $CLERR,ERRDTE,,.ERROR
202
203

```

```

150 .SBTTL $TMUNT - SETUP UNIT TEMPLATE
151 ;+
152 *** - $TMUNT - SETUP UNIT TEMPLATE
153 ;
154 INPUT:
155 R3-R5 - TPARS REGISTERS
156 ;
157 OUTPUT:
158 C-BIT = SUCCESS/FAILURE
159 IF SUCCESS, THE UNIT TEMPLATE IS STORED IN THE END OF TASK BUFFER.
160 ;
161 -
162
163 000002
164 000002 010546
165 000004 012701 000026
166 000010 004767 000000G
167 000014 010067 000000G
168 000020 005720
169 000022 010067 000000G
170 000026 012710 000000G
171 000032 005001
172 000034 012702 000000'
173 000040 012705 000000'
174 000044 004767 000000G
175 000050 103002
176 000052 005267 000000G
177 000056 012605
178 000060 000207

 $TMUNT::
 MOV R5, -(SP) ; SAVE R5
 MOV #C.LEN, R1 ; GET LENGTH OF ALLOCATION
 CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
 MOV R0, TEMP ; STORE TEMPLATE ADDRESS
 TST (R0)+ ; POINT TO C.SIS
 MOV R0, .FLAGS ; FOR OPTION SETTING
 MOV #CS.UNT, (R0) ; INDICATE THAT THIS IS A UNT$DF TEMPLATE
 CLR R1 ; IGNORE BLANKS
 MOV #UNTKTB, R2 ; KEYWORD TABLE
 MOV #UNTTTB, R5 ; STATE TABLE
 CALL .TPARS ; PARSE THE REST OF THE LINE
 BCC 20$; NORMAL RETURN IF NO ERROR
 INC $ERROR ; INDICATE SYNTAX ERROR
 MOV (SP)+, R5 ; RESTORE R5
 20$: RETURN

```

.TITLE TMPX2P - PARSE X.25 LEVEL 2 PROTOCOL DEFINITIONS  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

# MODULE DESCRIPTION:

STATE TABLE PARSE X.25 LEVEL 2 PROTOCOL DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

## IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0

TMPIX29 - PARSE X.29 CHARACTERIS MACRU V05.03b Saturday 29-Jun-85 05:46 E 11  
Table of contents

|    |     |                               |
|----|-----|-------------------------------|
| 4- | 57  | MACRO DEFINITIONS             |
| 5- | 126 | LOCAL DATA                    |
| 6- | 138 | TPARS STATE TABLES            |
| 7- | 167 | \$TMX29 - BUILD X29 TEMPLATE  |
| 8- | 201 | SETMXC - SET MAXIMUM CIRCUITS |
| 8- | 202 | SETCT - SET CALL TIMER VALUE  |

.TITLE TMPX3P - PARSE X.25 LEVEL 3 CHARACTERISTICS  
.IDENT /V05.00/  
.ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE X.25 LEVEL 3 CHARACTERISTICS

#### DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-AUG-81  
DECNET-11M/S V3.1  
DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/R SX V1.0



UPDBUF - CFE UPDATE BUFFER DEF1 MACRO V05.03b Saturday 29-Jun-85 <sup>F.13</sup> 05:47  
Table of contents

6- 174    \$UPBUF - UPDATE BUFFER TEMPLATE  
7- 220    SETNUM - SET THE NEXT NUMBER

```

57 .SBTTL MACRO CALLS AND LOCAL DATA
58 .SBTTL MACRO DEFINITIONS
59 ;
60 ; LOCAL MACROS
61 ;
62 ;
63 .MACRO PRINT TEXT
64 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
65 CALL $PRINT ; PRINT MESSAGE
66 .ENDM PRINT
67 ;
68 .MACRO ERRORS$ TEXT
69 ;
70 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
71 ; MESSAGE STRING.
72 ;
73 .IF DIF <TEXT><R0>
74 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
75 .ENDC
76 CALL $CFERR ; PRINT ERROR MESSAGE
77 .ENDM ERRORS$
78 ;
79 .MACRO EPRINT TEXT
80 ;
81 ; PRINT TEXT ON ERROR LUN
82 ;
83 .IF DIF <TEXT><R0>
84 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
85 .ENDC
86 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
87 .ENDM EPRINT
88 ;
89 ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
90 ;
91 .MACRO $$$PTN ROU
92 .MACRO 'ROU OFS,OCTL,OPT,?L
93 .IF NB <OPT>
94 MOV OPT,R2
95 CALL $PTOPT
96 BCS L
97 .IFTF
98 MOV OFS,R2
99 .IF NB <OCTL>
100 MOV SP,R1
101 .IFF
102 CLR R1
103 .ENDC ; IF NB <OCTL>
104 CALL 'ROU
105 .IFT ; IF NB <OPT>
106 L:
107 .ENDC ; IF NB <OPT>
108 .ENDM 'ROU
109 .ENDM $$$PTN
110
111 $$$PTN $PTNUM
112 $$$PTN $PTNMB
113 ;

```

```

181 .SBTTL SETCNT - CHECK CONTROLLER NUMBER
182 ;+
183 *** - SETCNT - CHECK CONTROLLER NUMBER
184 ;
185 INPUT:
186 .PNUMB = CONTROLLER NUMBER
187 .PNUMH = (HIGH ORDER)
188 ;
189 OUTPUT:
190 C-BIT = SUCCESS/FAILURE
191 ;
192 :-
193
194 000114 SETCNT: CALL $MOVPT ; MOVE THE CONTROLLER NUMBER INTO THE OUTPUT BUFFER
195 000114 004767 000000G MOV TEMP,RO ; GET TEMPLATE ADDRESS
196 000120 016700 177654 CMP .PNUMB,C.CNT(RO) ; IS THIS THE RIGHT CONTROLLER TEMPLATE ?
197 000124 026760 000000G 000004 BEQ 10$; IF EQ, YES - OKAY
198 000132 001402 BEQ 10$; IF EQ, YES - OKAY
199 000134 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
200 000140 000207 10$: RETURN

```

```

131 .SBTTL LOCAL DATA
132
133 ;
134 ; DEFINE THE CUG DESCRIPTOR BLOCK
135 ;
136 .ASECT
137 .=0
138 .BLKW 1 ; C.LNK
139 .BLKW 1 ; C.STS
140 .BLKW 2 ; CUG NAME
141 .BLKW 1 ; GROUP NUMBER
142 .BLKW 1 ; FLAGS
143 .LEN= ; LENGTH OF THE BLOCK
144 .PSECT
145 .NLIST BEX
146 .ASCII <11>/CUG$DF/<11>
147 CUGLN=-CUGDF
148 .LIST BEX
149 .EVEN
150
000000 000000
000000
000002
000004
000010
000012
000014
000000
000010 011 103 125 CUGDF:
000010

```

```

138 .SBTTL TPARS STATE TABLES
139 ;
140 ; INITIALIZE TABLES
141 ;
142 ;
143 ISTAT$ RDTSTB, RDTKTB
144 ;
145 ; PROCESS 'RDT$DF'
146 ;
147 STATES ; DTE NAME
148 TRANS $LAMDA,, $CLERR,ERRNAM,.ERROR
149 STATES
150 TRANS $STRNG,,SETNAM
151 STATES
152 TRANS <','>
153 ;
154 STATES ; DTE ADDRESS
155 TRANS $LAMDA,, $CLERR,ERRADR,.ERROR
156 STATES
157 TRANS !DTE,,SETADR
158 STATES
159 TRANS $EOS,$EXIT
160 ;
161 ; SUBEXPRESSION TO PARSE DTE STRING
162 ;
163 STATES DTE
164 TRANS $DIGIT
165 STATES DTE1
166 TRANS $DIGIT,DTE1
167 TRANS $LAMDA,$EXIT
168 ;
169 ; FINAL STATE
170 ;
171 STATES
172

```

```

195 .SBTTL SETNAM - SETUP REMOTE NODE NAME
196
197 +
198 *** - SETNAM - SETUP REMOTE NODE NAME
199
200 INPUT:
201 .PSTPT = THE ADDRESS OF THE NODE NAME STRING
202 .PSTCN = THE LENGTH OF THE NODE NAME STRING
203 TEMP = TEMPLATE ADDRESS
204
205 OUTPUT:
206 THE NODE NAME IS MOVED INTO C.NAM OF THE TEMPLATE
207 C-BIT = SUCCESS/FAILURE
208 -
209
210 SETNAM:
211 000202 016700 000000G MOV .PSTPT,R0 ; POINT AT THE NODE NAME STRING
212 000206 005001 CLR R1 ; PERIODS ARE NOT ALLOWED
213 000210 004767 000000G CALL $CAT5 ; CONVERT NODE NAME TO RAD50
214 000214 016702 177560 MOV TEMP,R2 ; POINT AT TEMPLATE
215 000220 010162 000004 MOV R1,C.NAM(R2) ; STORE THE NODE NAME
216 000224 103415 BCS 10$; IF CS, LESS THAN 3 CHARS CONVERTED
217 000226 005001 CLR R1 ; PERIODS NOT ALLOWED
218 000230 004767 000000G CALL $CAT5 ; CONVERT NODE NAME TO RAD50
219 000234 016702 177540 MOV TEMP,R2 ; POINT AT TEMPLATE
220 000240 010162 000006 MOV R1,C.NAM+2(R2) ; STORE THE 2ND HALF OF THE NODE NAME
221 000244 022767 000006 000000G CMP #6,.PSTCN ; IS THE NODE NAME TOO BIG ?
222 000252 103002 BHIS 10$; IF HIS, NO .. OKAY
223 000254 062716 000002 ADD #2,(SP) ; ELSE, REJECT TRANSITION
224 000260 000207 10$: RETURN

```

```

208 .SBTTL $TMROU - SETUP ROUTING TEMPLATE
209 +
210 *** - $TMROU - SETUP ROUTING TEMPLATE
211 :
212 INPUT:
213 R3-R5 - TPARS REGISTERS
214 :
215 OUTPUT:
216 C-BIT = SUCCESS/FAILURE
217 IF SUCCESS, THE ROUTING TEMPLATE IS STORED IN THE END OF TASK BUFFER.
218 :
219 -
220
221 $TMROU:
222 000002 010546 MOV R5,-(SP) ; SAVE R5
223 000004 012701 MOV #C.LEN,R1 ; GET LENGTH OF NEEDED ALLOCATION
224 000010 004767 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
225 000014 010067 MOV R0,.TEMP ; SAVE TEMPLATE ADDRESS
226 000020 005720 TST (R0)+ ; POINT TO C.STS
227 000022 010067 MOV R0,.FLAGS ; SAVE FOR OPTION SETTING
228 000026 012710 MOV #C$.ROU,(R0) ; INDICATE THAT THIS IS THE ROU$DF TEMPLATE
229 000032 005067 CLR .ERROR ; NO ERROR DETAIL YET
230 000036 005001 CLR R1 ; IGNORE BLANKS
231 000040 012702 MOV #ROUKTB,R2 ; KEYWORD TABLE
232 000044 012705 MOV #ROUSTB,R5 ; STATE TABLE
233 000050 004767 CALL .TPARS ; PARSE THE REST OF THE LINE
234 000054 103012 BCC 20$; NORMAL RETURN IF NO ERROR
235 000056 016700 MOV .ERROR,R0 ; CHECK FOR ERROR DETAIL TEXT
236 000062 001405 BEQ 15$; IF EQ, NO DETAIL TEXT
237 000064 ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
238 000070 012767 MOV #-2,$ERROR ; INDICATE MESSAGE PRINTED
239 000076 005267 INC $ERROR ; INDICATE SYNTAX ERROR
240 000102 012605 MOV (SP)+,R5 ; RESTORE R5
241 000104 000207 RETURN
242

```

```

222 .SBTTL $TMSE - SETUP REMOTE CHARACTERISTICS TEMPLATE
223
224 ;+
225 ;*** - $TMSE - SETUP REMOTE CHARACTERISTICS TEMPLATE
226
227 INPUT:
228 R3-R5 - TPARS REGISTERS
229
230 OUTPUT:
231 C-BIT = SUCCESS/FAILURE
232 IF SUCCESS, THE REMOTE TEMPLATE IS STORED IN THE END OF TASK BUFFER.
233
234 ;-
235
236 $TMSE::
237 000022 MOV R5, -(SP) ; SAVE R5
238 000022 C'IR SERFLG ; INITIALIZE FLAGS
239 000024 010546 177750 MOV #$$LEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
240 000030 012701 000056 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
241 000034 004767 000000G MOV RO, TEMP ; SAVE TEMPLATE ADDRESS
242 000040 010067 000000G TST (RO)+ ; POINT TO C.STS
243 000044 005720 MOV RO, FLAGS ; SAVE FOR OPTION SETTING
244 000046 010067 000000G MOV #CS.SER, (RO) ; INDICATE THAT THIS IS A DST$DF TEMPLATE
245 000052 012710 000000G ADD #S.LENX, $BYTMN ; UPDATE MINIMUM ALLOCATION
246 000056 062767 000000G 000000G CLR .ERROR ; NO ERROR DETAIL YET
247 000064 005067 000000G CLR R1 ; IGNORE BLANKS
248 000070 005001 MOV #SERKTB, R2 ; KEYWORD TABLE
249 000072 012702 000000' MOV #SERSTB, R5 ; STATE TABLE
250 000076 012705 000000' CALL .TPARS ; PARSE THE REST OF THE LINE
251 000102 004767 000000G MOV .TEMP, RO ; POINT TO TEMPLATE
252 000106 016700 000000G MOV SERFLG, $$FLG(RO) ; SET PARSED OPTIONS
253 000112 016760 177662 000004 BCC 20$; NORMAL RETURN IF NO ERROR
254 000120 103005 MOV #2, $ERROR ; SUPPRESS SYNTAX MESSAGE
255 000122 012767 177776 000000G 15$: INC $ERROR ; INDICATE SYNTAX ERROR
256 000130 005267 000000G 20$: MOV (SP)+, R5 ; RESTORE R5
257 000134 012605 RETURN
258 000136 000207

```



.TITLE TMPSLT - CFE PARSE LINE DEFINITION IN CETAB  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE LINE DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
 DECNET-11M/S V3.0  
 DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RX V1.0

|          |          |     |         |          |          |          |          |          |          |            |          |
|----------|----------|-----|---------|----------|----------|----------|----------|----------|----------|------------|----------|
| BITS     | 000266R  | 002 | LF.BRO= | ***** GX | SETRPR   | 000652R  | \$SLNAM= | ***** GX | \$GTR50= | ***** GX   |          |
| CS.SLT=  | ***** GX |     | LF.ENA= | ***** GX | SETSLV   | 000734R  | \$SLRPR= | ***** GX | \$LAMDA= | 000000     |          |
| CTI      | 000142R  | 002 | LF.MDC= | ***** GX | SETUNT   | 000556R  | \$SLTYP= | ***** GX | \$NUMBR= | 000002     |          |
| C.STS =  | ***** GX |     | LF.MFL= | ***** GX | SKIP     | 000252R  | \$SLUNT= | ***** GX | \$RAD50= | 000016     |          |
| ERRCNT   | 000147R  |     | LF.MTP= | ***** GX | SLTKTB   | 000000RG | \$ALPHA= | 000022   | \$STAN = | ***** GX   |          |
| ERRCTT   | 000337R  |     | LF.SER= | ***** GX | SLTSTB   | 000000RG | \$ANY =  | 000020   | \$STRNG= | 000004     |          |
| ERRDLC   | 000034R  |     | LF.TIM= | ***** GX | SL\$BRA= | ***** GX | \$BLANK= | 000006   | \$SUBXP= | 000010     |          |
| ERRFLG   | 000120R  |     | LF.X2P= | ***** GX | SL\$CTM= | ***** GX | \$CFERR= | ***** GX | \$TALOC= | ***** GX   |          |
| ERRHTM   | 000302R  |     | NBRA    | 000212R  | 002      | SL\$HTM= | ***** GX | \$CLERR= | ***** GX | \$TMSLT    | 000376RG |
| ERRLLC   | 000066R  |     | NEXT    | 000262R  | 002      | SL\$RPR= | ***** GX | \$DIGIT= | 000024   | \$\$\$FLG= | 177777   |
| ERRNAM   | 000002R  |     | RPR1    | 000230R  | 002      | SL\$TYP= | ***** GX | \$DNUMB= | 000014   | \$\$\$KEY= | 000012   |
| ERRTYP   | 000247R  |     | SETBRA  | 000700R  |          | SL\$BRA= | ***** GX | \$EOS =  | 000012   | \$\$\$STA= | 000000   |
| ERRUNT   | 000212R  |     | SETCNT  | 000536R  |          | SL\$CNT= | ***** GX | \$ERROR= | ***** GX | \$\$\$TMP= | 000054R  |
| EXIT     | 000366R  | 002 | SETCTT  | 000576R  |          | SL\$CTT= | ***** GX | \$EXIT = | 000000   | .ERROR=    | ***** GX |
| EX\$SEV= | 000004   |     | SETFLG  | 000522R  |          | SL\$FLG= | ***** GX | \$FAIL = | 177777   | .FLAGS=    | ***** GX |
| FLAG     | 000000R  |     | SETHTM  | 000624R  |          | SL\$HTM= | ***** GX | \$GTNMB= | ***** GX | .TEMP =    | ***** GX |
| HTM      | 000166R  | 002 | SETMST  | 000726R  |          | \$LLEN=  | ***** GX | \$GTNUM= | ***** GX | .TPARS=    | ***** GX |
| LFS      | 000244R  | 002 | SETNAM  | 000506R  |          |          |          |          |          |            |          |

  

|         |        |     |                    |
|---------|--------|-----|--------------------|
| . ABS.  | 000000 | 000 | (RW,I,GBL,ABS,OVR) |
|         | 000756 | 001 | (RW,I,LCL,REL,CON) |
| \$STATE | 000374 | 002 | (RW,D,LCL,REL,CON) |
| \$KTAB  | 000026 | 003 | (RW,D,LCL,REL,CON) |
| \$KSTR  | 000060 | 004 | (RW,D,LCL,REL,CON) |

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10928 Words ( 43 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:54.21  
SY:TMPSLT.V2,[132,134]TMPSLT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPSLT

TMPSTA - CFE PARSE STATION DEFIN MACRO V05.03b Saturday 29-Jun-85 05:43 Page 8  
 SETCST - SETUP STATION COST AND FLAGS

```

248 .SBTTL SETCST - SETUP STATION COST AND FLAGS
249
250 +
251 *** - SETCST - SETUP STATION COST
252
253 THIS ACTION ROUTINE SETS UP THE STATION COST
254
255 INPUT:
256 .PNUMB = STATION COST
257 .PNUMH = (HIGH ORDER)
258
259 OUTPUT:
260 C.CST = STATION COST
261 C.FLG = STATION FLAGS
262
263 -
264
265 SETCST:
266 MOV TEMP,RO ; GET TEMPLATE COST
267 MOV .PNUMB,C.CST(RO) ; STORE THE STATION COST IN TEMPLATE
268 CLR C.FLG(RO) ; CLEAR THE STATION FLAGS FIELD
269 BIS FLAG,C.FLG(RO) ; SETUP THE STATION FLAGS
270 TST .PNUMH ; IS IT A WORD VALUE ?
271 BNE 10$; IF NE, NO - REJECT
272 CMP .PNUMB,#MXCSTL ; IS THE STATION COST IN RANGE ?
273 BLOS 20$; IF LOS, YES - OKAY
274 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
275 RETURN

```

TMPSTA - CFE PARSE STATION DEFIN MACRO V05.03b Saturday 29-Jun-85 05:43 Page 9  
 SETCST - SETUP STATION COST AND FLAGS

```

204 000336 STATES$ DTE,OPT1,SETDTE ; DTE OPTIONAL ADDRESS
205 000336 TRANS$ $LAMBDA
206 000336 TRANS$ $LAMBDA
207 000336 STATES$ OPT1
208 000336 TRANS$ <'>,, $CLERR,ERROWN,.ERROR
209
210 000336 STATES$; OWNER PROCESS
211 000336 TRANS$ $STRNG,,SETOWN
212
213 000336 STATES$
214 000336 TRANS$ $EOS,$EXIT
215
216
217 : SUBEXPRESSION TO PARSE DTE ADDRESS STRING
218 :
219 000336 STATES$ DTE
220 000336 TRANS$ $DIGIT
221 000336 STATES$ DTE1
222 000336 TRANS$ $DIGIT,DTE1
223 000336 TRANS$ $LAMBDA,$EXIT
224
225
226 : FINAL STATE
227 :
228 000336 STATES$
229

```

```

180 .SBTTL SETUNT - SET UP UNIT NUMBER
181 ;+
182 *** - SETUNT - SETUP UNIT NUMBER
183 ;
184 THIS ACTION ROUTINE SETS UP THE UNIT NUMBER
185 ;
186 INPUT:
187 .PNUMB = UNIT NUMBER
188 .PNUMH = (HIGH ORDER)
189 .TEMP = TEMPLATE ADDRESS
190 ;
191 OUTPUT:
192 C.UNT = UNIT NUMBER
193 ;
194 -
195
196 000062 SETUNT:
197 000062 016700 000000G MOV .TEMP,R0 ; GET TEMPLATE ADDRESS
198 000066 016760 000000G 000004 MOV .PNUMB,C.UNT(R0) ; STORE THE UNIT NUMBER IN TEMPLATE
199 000074 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
200 000100 001003 BNE 10$; IF NE, NO - REJECT
201 000102 105767 000001G TSTB .PNUMB+1 ; IS THE UNIT NUMBER IN RANGE ?
202 000106 001402 BEQ 20$; IF EQ, YES - OKAY
203 000110 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
204 000114 000207 20$: RETURN

```

```

57 .SBTTL MACRO DEFINITIONS
58 ;
59 ; LOCAL MACROS
60 ;
61 ;
62 .MACRO PRINT TEXT
63 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM PRINT
66 ;
67 .MACRO ERRORS$ TEXT
68 ;
69 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
70 ; MESSAGE STRING.
71 ;
72 .IF DIF <TEXT><R0>
73 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
74 .ENDC
75 CALL $CFERR ; PRINT ERROR MESSAGE
76 .ENDM ERRORS$
77 ;
78 .MACRO EPRINT TEXT
79 ;
80 ; PRINT TEXT ON ERROR LUN
81 ;
82 .IF DIF <TEXT><R0>
83 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
84 .ENDC
85 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
86 .ENDM EPRINT
87 ;
88 ; GET NUMBER WITH RANGE CHECKING
89 ;
90 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
91 MOV OFFS,R2
92 MOV LO,R0
93 MOV HI,R1
94 .IF NB <OPT>
95 BIS OPT,@.FLAGS
96 .ENDC
97 JMP $GTNUM
98 .ENDM
99 ;
100 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#177777>,OPT
101 MOV OFFS,R2
102 MOV LO,R0
103 MOV HI,R1
104 .IF NB <OPT>
105 BIS OPT,@.FLAGS
106 .ENDC
107 JMP $GTNMB
108 .ENDM
109 ;
110 ; GET RAD50 WITH FIELD SIZE CHECKING
111 ;
112 .MACRO $GTR50 OFFS,LEN,OPT
113 MOV OFFS,R2

```

.TITLE TMPX29 - PARSE X.29 CHARACTERISTICS  
.IDENT /V05.00/  
.ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE X.29 CHARACTERISTICS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-AUG-81  
DECNET-11M/S V3.1  
DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/Rsx V1.0

:

```

57 .SBTTL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61 .MACRO PRINT TEXT
62 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
63 CALL $PRINT ; PRINT MESSAGE
64 .ENDM PRINT
65
66 .MACRO ERROR$ TEXT
67
68 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
69 ; MESSAGE STRING.
70
71 .IF DIF <TEXT><R0>
72 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
73 .ENDC
74 CALL $CFERR ; PRINT ERROR MESSAGE
75 .ENDM ERROR$
76
77 .MACRO EPRINT TEXT
78
79 ; PRINT TEXT ON ERROR LUN
80
81 .IF DIF <TEXT><R0>
82 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
83 .ENDC
84 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
85 .ENDM EPRINT
86
87 ; GET NUMBER WITH RANGE CHECKING
88
89 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#17777>,OPT
90 MOV OFFS,R2
91 MOV LO,R0
92 MOV HI,R1
93 .IF NB <OPT>
94 BIS OPT,@.FLAGS
95 .ENDC
96 JMP $GTNUM
97 .ENDM
98
99 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
100 MOV OFFS,R2
101 MOV LO,R0
102 MOV HI,R1
103 .IF NB <OPT>
104 BIS OPT,@.FLAGS
105 .ENDC
106 JMP $GTNMB
107 .ENDM
108
109 ; GET RAD50 WITH FIELD SIZE CHECKING
110
111 .MACRO $GTR50 OFFS,LEN,OPT
112 MOV OFFS,R2
113

```



.TITLE UPDBUF - CFE UPDATE BUFFER DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

# MODULE DESCRIPTION:

CFE - UPDATE STATE TABLE TO PARSE THE BUFFER DEFINITION

## DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/Rsx V1.0

```
114 ; PUT RAD50 TEMPLATE FIELD TO BUFFER
115 ;
116 ; MACRO $PTR50 OFS,WC
117 MOV OFS,R2
118 MOV WC,R1
119 CALL $PTR50
120 .ENDM
121
```

```

202 .SBTTL SETVE - SETUP THE VECTOR ADDRESS
203 +
204 *** - SETVEC - SETUP THE VECTOR ADDRESS
205 :
206 INPUT:
207 .PNUMB = THE VECTOR ADDRESS
208 .PNUMH = (HIGH ORDER)
209 TEMP = START ADDRESS OF THE TEMPLATE
210 :
211 OUTPUT:
212 :
213 -
214
215 SETVEC: MOV TEMP,R2 ; GET START OF TEMPLATE
216 000142 016702 177632 MOV C_VEC(R2),R1 ; GET THE VECTOR ADDRESS
217 000146 016201 000006 MOV $OBUF,R0 ; GET THE CURRENT POSITION IN THE OUTPUT BUFFER
218 000152 016700 000000G CALL $CNVO ; CONVERT NUMBER TO OCTAL
219 000156 004767 000000G MOV R0,$OBUF ; UPDATE THE POINTER INTO THE OUTPUT BUFFER
220 000162 010067 000000G
221 000166 000207

```

```

152 .SBTTL $UPCUG - UPDATE PERMANENT VIRTUAL CIRCUIT
153
154 ;+
155 *** - $UPCUG - UPDATE PERMANENT VIRTUAL CIRCUIT
156 :
157 INPUT:
158 R5 - TEMPLATE ADDRESS
159 :
160 OUTPUT:
161 C-BIT = SUCCESS/FAILURE
162 :
163 -
164 000010 $UPCUG::
165 000010 012767 000000G 000000C MOV #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
166 000016 010567 000000G MOV R5,TEMP ; SAVE TEMPLATE ADDRESS
167 000022 005725 TST (R5)+ ; POINT TO C.STS
168 000024 010567 000000G MOV R5,FLAGS ; SAVE ADDRESS OF FLAGS WORD
169 000030 012767 177777 000000G MOV #1,COMMA ; INITIALIZE FOR OPTIONS
170 000036 012700 000000G MOV #.BUFF,R0 ; POINT AT START OF OUTPUT BUFFER
171 000042 012701 000000 MOV #CUGDF,R1 ; POINT AT START OF CUG DEFINITION
172 000046 012702 000010 MOV #CUGLN,R2 ; GET LENGTH OF CUG DEFINITION
173 000052 004767 000000G CALL $MVASC ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
174 000056 010067 000000G MOV R0,$OBUF ; SAVE CURRENT POSITION IN OUTPUT BUFFER
175 000062 $PTR50 #C.NAM,#2 ; CUG NAME
176 000076 $PTNUM #C.CUG ; GROUP
177 000110 005367 000000G DEC $OBUF ; BACK OVER DECIMAL POINT
178 000114 $PTNMB #C.FLG,OCTAL,#CS.FLG ; OPTIONAL FLAGS
179 000140 016701 000000G MOV $OBUF,R1 ; CALCULATE LENGTH OF BUFFER
180 000144 162701 000000G SUB #.BUFF,R1
181 000150 010167 000000C MOV R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
182 000154 004767 000000G CALL $WDATA ; WRITE THE RECORD TO THE OUTPUT FILE
183 000160 012767 000000G 000000C MOV #$BUF1,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE "END$DF" LINE
184 000166 000207 RETURN
185
186 000001 .END

```

```

174 .SBTTL $TMRDT - BUILD RDT TEMPLATE
175 ;+
176 ;*** - $TMRDT - BUILD RDT TEMPLATE
177 ;
178 ; THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
179 ;
180 ; INPUT:
181 ; R3-R5 - TPARS REGISTERS
182 ;
183 ; OUTPUT:
184 ; C-BIT = SUCCESS/FAILURE
185 ; THE RDT TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
186 ;
187 ; -
188
189 000066 $TMRDT::
190 000066 MOV R5, -(SP) ; SAVE R5
191 000070 MOV #R$DLEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
192 000074 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
193 000100 MOV R0, TEMP ; SAVE TEMPLATE ADDRESS
194 000104 TST (R0)+ ; POINT TO C.STS
195 000106 MOV R0, .FLAGS ; SAVE FOR OPTION SETTING
196 000112 MOV #CS.RDT, (R0) ; INDICATE THAT THIS IS A RDT$DF TEMPLATE
197 000116 CLR .ERROR ; NO ERROR DETAIL YET
198 000122 ADD #R$LENX, $BYTMN ; UPDATE MINIMUM POOL BYTE EXTENSION
199 000130 CLR R1 ; IGNORE BLANKS
200 000132 MOV #RDTKTIB, R2 ; KEYWORD TABLE
201 000136 MOV #RDTSTIB, R5 ; STATE TABLE
202 000142 CALL .TPARS ; PARSE THE REST OF THE LINE
203 000146 BCC 20$; NORMAL RETURN IF NO ERROR
204 000150 MOV .ERROR, R0 ; CHECK FOR ERROR DETAIL TEXT
205 000154 BEQ 15$; IF EQ, NO DETAIL TEXT
206 000156 MOV ERROR$, R0 ; PRINT ACTION ROUTINE ERROR
207 000162 MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
208 000170 INC $ERROR ; INDICATE SYNTAX ERROR
209 000174 MOV (SP)+, R5 ; RESTORE R5
210 000176 RETURN

```

```

226 .SBTTL SETARE - SET UP REMOTE AREA
227
228 ;+
229 ;*** - SETARE - SETUP REMOTE AREA
230
231 ; INPUT:
232 ; .PNUMB = THE REMOTE ADDRESS
233 ; TEMP = TEMPLATE ADDRESS
234
235 ; OUTPUT:
236 ; C.ADD = THE REMOTE NODE AREA IN UPPER 6 BITS
237 ; -
238
239 SETARE:
240 MOV .PNUMB,R0 ; PICK UP PARSED VALUE
241 BEQ 10$; OH COME ON, BE SERIOUS !
242 CMP R0,#63. ; COMPARE HIGH END OF RANGE
243 BHI 10$; IF HI, N.G.
244
245 5$: MOV TEMP,R1 ; GET THE REMOTE TEMPLATE
246 SWAB R0 ; GET AREA NUMBER INTO HIGH BYTE
247 ASL R0 ; WANT IT IN THE ...
248 ASL R0 ; ... TOP 6 BITS
249 MOV R0,C.ADD(R1) ; SET AREA, INITIALIZE ADDRESS TO ZERO
250 BR 20$; ALL DONE
251 10$: ADD #2,(SP) ; ELSE, REJECT TRANSITION
252 20$: RETURN
253
240 000262 016700 000000G
241 000266 001413
242 000270 020027 000077
243 000274 101010
245 000276 016701 177476
246 000302 000300
247 000304 006300
248 000306 006300
249 000310 010061 000010
250 000314 000402
251 000316 062716 000002
252 000322 000207

```

```

244
245
246 000106 SETADD: $GTNUM #C.ADD,,#1023.
247 000106
248
249 000126 SETCST: $GTNUM #C.CST,,#MAXCST
250 000126
251
252 000146 SETHOP: $GTNUM #C.HOP,,#MAXHOP
253 000146
254
255 000166 SETRTM: $GTNUM #C.RTM,,#MAXRTM
256 000166
257
258 000206 SETNA: $GTNUM #C.NA
259 000206
260
261 000226 SENBEA: $GTNUM #C.NBEA
262 000226
263
264 000246 SEAHOP: $GTNUM #C.AHOP
265 000246
266
267 000266 SEACST: $GTNUM #C.ACST
268 000266
269
270 000306 SELV2T: $GTNUM #C.L2TM
271 000306
272
273
274 000001 .END

```

```

259 .SBTTL SETADD - SET UP REMOTE ADDRESS
260 .SBTTL HOSADD - SET UP HOST ADDRESS
261
262 ;+
263 *** - SETADD - SETUP REMOTE ADDRESS
264 *** - HOSADD - SETUP HOST ADDRESS
265
266 INPUT:
267 .PNUMB = THE REMOTE ADDRESS
268 .TEMP = TEMPLATE ADDRESS
269
270 OUTPUT:
271 $$ADD = THE REMOTE NODE ADDRESS
272 $$HST = THE HOST NODE ADDRESS
273
274 .ENABL LSB
275
276 HOSADD: MOV #$$HST,R1 ; SET UP FOR HOST
277 BR 5$; JOIN COMMON CODE
278
279 SETADD: MOV #$$ADD,R1 ; SET UP FOR REMOTE
280 MOV .PNUMB,R0 ; PICK UP PARSED VALUE
281 BEQ REJ ; OH COME ON, BE SERIOUS !
282 CMP R0,#1023. ; COMPARE HIGH END OF RANGE
283 BHI REJ ; IF HI, NG
284 ADD .TEMP,R1 ; GET THE REMOTE TEMPLATE
285 BIS R0,R1) ; FOLD ADDRESS IN WITH AREA
286
287 20$: RETURN
288
289 .DSABL LSB
290
291 .SBTTL SETARE - SET UP REMOTE AREA
292
293 ;+
294 *** - SETARE - SETUP REMOTE AREA
295
296 INPUT:
297 .PNUMB = THE REMOTE ADDRESS
298 .TEMP = TEMPLATE ADDRESS
299
300 OUTPUT:
301 $$ADD = THE REMOTE NODE ADDRESS
302 $$HST = THE HOST NODE ADDRESS
303
304 .ENABL LSB
305
306 HOSARE: MOV #$$HST,R1 ; SET UP FOR HOST
307 BR 2$; JOIN COMMON CODE
308
309 SETARE: MOV #$$ADD,R1 ; SET UP FOR REMOTE
310 MOV .PNUMB,R0 ; PICK UP PARSED VALUE
311 BEQ 10$; OH COME ON, BE SERIOUS !
312 CMP R0,#63. ; COMPARE HIGH END OF RANGE
313 BHI 10$; IF HI, N.G.
314
315 5$: ADD .TEMP,R1 ; GET THE REMOTE TEMPLATE
316 SWAB R0 ; GET AREA NUMBER INTO HIGH BYTE
317 ASL R0 ; WANT IT IN THE ...
318 ASL R0 ; ... TOP 6 BITS

```



```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM
63
64 .MACRO ERRORS$ TEXT
65
66 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
67 ; MESSAGE STRING.
68
69 .IF DIF <TEXT><R0>
70 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
71 .ENDC
72 CALL $CFERR ; PRINT ERROR MESSAGE
73 .ENDM ERRORS$
74
75
76 ; GET NUMBER WITH RANGE CHECKING
77
78 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
79 MOV OFFS,R2
80 MOV LO,R0
81 MOV HI,R1
82 .IF NB <OPT>
83 BIS OPT,@.FLAGS
84 .ENDC
85 JMP $GTNUM
86 .ENDM
87
88 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#177777>,OPT
89 MOV OFFS,R2
90 MOV LO,R0
91 MOV HI,R1
92 .IF NB <OPT>
93 BIS OPT,@.FLAGS
94 .ENDC
95 JMP $GTNMB
96 .ENDM
97
98 ; GET RAD50 WITH FIELD SIZE CHECKING
99
100 .MACRO $GTR50 OFFS,LEN,OPT
101 MOV OFFS,R2
102 MOV LEN,R1
103 .IF NB <OPT>
104 BIS OPT,@.FLAGS
105 .ENDC
106 JMP $GTR50
107 .ENDM
108
109 ;***
110 ; MACRO CALLS
111 ;****

```

TEMPLT      CREATED BY    MACRO    ON 29-JUN-85 AT 05:43      PAGE 1      G 6  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL   | VALUE      | REFERENCES                                                                                     |
|----------|------------|------------------------------------------------------------------------------------------------|
| CS.SLT   | = ***** GX | 6-272                                                                                          |
| C.STS    | = ***** GX | *13-405                                                                                        |
| ERRCNT   | 000147 R   | #4-128                                                                                         |
| ERRCTT   | 000337 R   | #4-132                                                                                         |
| ERRDLC   | 000034 R   | #4-125                                                                                         |
| ERRFLG   | 000120 R   | #4-127                                                                                         |
| ERRHTM   | 000302 R   | #4-131                                                                                         |
| ERRLLC   | 000066 R   | #4-126                                                                                         |
| ERRNAM   | 000002 R   | #4-124                                                                                         |
| ERRTYP   | 000247 R   | #4-130                                                                                         |
| ERRUNT   | 000212 R   | #4-129                                                                                         |
| EX\$SEV  | = 000004   | #4-122      4-124      4-125      4-126      4-127      4-128      4-129      4-130      4-131 |
| FLAG     | 000000 R   | #4-118      *6-266      8-318                                                                  |
| SETBRA   | 000700 R   | #12-385                                                                                        |
| SETCNT   | 000536 R   | #9-334                                                                                         |
| SETCTT   | 000576 R   | #11-364                                                                                        |
| SETFLG   | 000522 R   | #8-316                                                                                         |
| SETHTM   | 000624 R   | #12-381                                                                                        |
| SETMST   | 000726 R   | #13-400                                                                                        |
| SETNAM   | 000506 R   | #7-301                                                                                         |
| SETRPR   | 000652 R   | #12-383                                                                                        |
| SETSLV   | 000734 R   | #13-403                                                                                        |
| SETUNT   | 000556 R   | #10-350                                                                                        |
| SLTKTB   | 000000 RG  | #5-140      6-274                                                                              |
| SLTSTB   | 000000 RG  | #5-140      6-275                                                                              |
| SL\$BRA  | = ***** GX | 12-385                                                                                         |
| SL\$CTM  | = ***** GX | 11-364                                                                                         |
| SL\$HTM  | = ***** GX | 12-381                                                                                         |
| SL\$RPR  | = ***** GX | 12-383                                                                                         |
| SL\$TYP  | = ***** GX | 13-405                                                                                         |
| SS\$BRA  | = ***** GX | 12-385                                                                                         |
| SS\$CNT  | = ***** GX | 9-334                                                                                          |
| SS\$CTT  | = ***** GX | 11-364                                                                                         |
| SS\$FLG  | = ***** GX | *8-318                                                                                         |
| SS\$LHTM | = ***** GX | 12-381                                                                                         |
| SS\$LEN  | = ***** GX | 6-267                                                                                          |
| SS\$NAM  | = ***** GX | 7-301                                                                                          |
| SS\$RPR  | = ***** GX | 12-383                                                                                         |
| SS\$TYP  | = ***** GX | *13-406                                                                                        |
| SS\$UNT  | = ***** GX | 10-350                                                                                         |
| \$ALPHA  | = 000022   | #5-140                                                                                         |
| \$ANY    | = 000020   | #5-140                                                                                         |
| \$BLANK  | = 000006   | #5-140                                                                                         |
| \$CFERR  | = ***** GX | 6-280                                                                                          |
| \$DIGIT  | = 000024   | #5-140                                                                                         |
| \$DNUMB  | = 000014   | #5-140                                                                                         |
| \$EOS    | = 000012   | #5-140                                                                                         |
| \$ERROR  | = ***** GX | *6-281      *6-282                                                                             |
| \$EXIT   | = 000000   | #5-140                                                                                         |
| \$FAIL   | = 177777   | #5-140                                                                                         |
| \$GPRM   | = *****    | 5-140                                                                                          |

```

276 .SBTTL SETAPR - SETUP STATION ACTIVE POLLING RATIO
277 ;+
278 *** - SETAPR - SETUP STATION ACTIVE POLLING RATIO
279 ;
280 THIS ACTION ROUTINE SETS UP THE STATION ACTIVE POLLING RATIO
281 ;
282 INPUT:
283 .PNUMB = STATION APR
284 .PNUMH = STATION APR (HIGH ORDER)
285 ;
286 OUTPUT:
287 C.APR = STATION APR
288 ;
289 -
290
291 000266 SETAPR:
292 000266 016700 177506 MOV TEMP,R0 ; GET TEMPLATE COST
293 000272 116760 000000G 000010 MOVB .PNUMB,C.APR(R0) ; STORE THE STATION APR IN TEMPLATE
294 000300 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
295 000304 001003 BNE 10$; IF NE, NO - REJECT
296 000306 105767 000001G TSTB .PNUMB+1 ; IS THE STATION APR IN RANGE ?
297 000312 001402 BEQ 20$; IF EQ, YES - OKAY
298 000314 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
299 000320 000207 20$: RETURN

```

```

231 .SBTTL $TMSVC - BUILD SVC TEMPLATE
232
233 +
234 *** - $TMSVC - BUILD SVC TEMPLATE
235
236 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
237
238 INPUT:
239 R3-R5 - TPARS REGISTERS
240
241 OUTPUT:
242 C-BIT = SUCCESS/FAILURE
243 THE SVC TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
244
245 -
246
247 $TMSVC::
248 CLR $SVUPD ; CLEAR UPDATE FLAG
249 MOV R5, -(SP) ; SAVE R5
250 MOV #S$VLEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
251 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
252 MOV R0, TEMP ; SAVE TEMPLATE ADDRESS
253 TST (R0)+ ; POINT TO C.STS
254 MOV R0, FLAGS ; SAVE FOR OPTION SETTING
255 MOV #CS.SVC, (R0) ; INDICATE THAT THIS IS A SVC$DF TEMPLATE
256 ADD #S$LENX, $BYTMN ; UPDATE MINIMUM BYTE ALLOCATION
257 CLR .ERROR ; NO ERROR DETAIL YET
258 CLR R1 ; IGNORE BLANKS
259 MOV #SVCKTB, R2 ; KEYWORD TABLE
260 MOV #SVCSTB, R5 ; STATE TABLE
261 CALL TPARS ; PARSE THE REST OF THE LINE
262 BCC 20$; NORMAL RETURN IF NO ERROR
263 MOV .ERROR, R0 ; CHECK FOR ERROR DETAIL TEXT
264 BEQ 15$; IF EQ, NO DETAIL TEXT
265 ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
266 MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
267 INC $ERROR ; INDICATE SYNTAX ERROR
268 MOV (SP)+, R5 ; RESTORE R5
269 RETURN

```

```

206 .SBTTL SETCWO - SETUP THE CHARACTERISTICS WORD 0
207 ;+
208 *** - SETCWO - SETUP THE CHARACTERISTICS WORD 0
209 ;
210 INPUT:
211 .PNUMB = THE CHARACTERISTICS WORD 0
212 .PNUMH = (HIGH ORDER)
213 .TEMP = START ADDRESS OF THE TEMPLATE
214 ;
215 OUTPUT:
216 THE CHARACTERISTICS WORD 0 IS MOVED INTO THE TEMPLATE
217 ;
218 ;-
219 ;
220 SETCWO:
221 000116 016700 000000G MOV .TEMP,R0 ; GET START OF TEMPLATE
222 000122 016760 000000G 000006 MOV .PNUMB,C.CWO(R0) ; STORE THE CHARACTERISTICS WORD 0
223 000130 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
224 000134 001402 BEQ 10$; IF EQ, YES
225 000136 062716 000002 ADD #2,(SP) ; ELSE, REJECT
226 000142 000207 10$: RETURN

```

```
114 MOV LEN,R1
115 .IF NB <OPT>
116 BIS OPT,a.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 ;
122 ; MACRO CALLS
123 ;
124 .MCALL ISTAT$,STATE$,TRANS$,CALLR
```

```

57 .SBTTL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61 .MACRO PRINT TEXT
62 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
63 CALL $PRINT ; PRINT MESSAGE
64 .ENDM PRINT
65
66 .MACRO ERROR$ TEXT
67
68 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
69 ; MESSAGE STRING.
70
71 .IF DIF <TEXT><R0>
72 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
73 .ENDC
74 CALL $CFERR ; PRINT ERROR MESSAGE
75 .ENDM ERROR$
76
77 .MACRO EPRINT TEXT
78
79 ; PRINT TEXT ON ERROR LUN
80
81 .IF DIF <TEXT><R0>
82 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
83 .ENDC
84 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
85 .ENDM EPRINT
86
87 ; GET NUMBER WITH RANGE CHECKING
88
89 .MACRO $GTNUM OFFS,LO=<#0>,HI=<#177777>,OPT
90 MOV OFFS,R2
91 MOV LO,R0
92 MOV HI,R1
93 .IF NB <OPT>
94 BIS OPT,a.FLAGS
95 .ENDC
96 JMP $GTNUM
97 .ENDM
98
99 .MACRO $GTNMB OFFS,LO=<#0>,HI=<#377>,OPT
100 MOV OFFS,R2
101 MOV LO,R0
102 MOV HI,R1
103 .IF NB <OPT>
104 BIS OPT,a.FLAGS
105 .ENDC
106 JMP $GTNMB
107 .ENDM
108
109 ; GET RAD50 WITH FIELD SIZE CHECKING
110
111 .MACRO $GTR50 OFFS,LEN,OPT
112 MOV OFFS,R2
113

```

```
114 MOV LEN,R1
115 .IF NB <OPT>
116 BIS OPT,a.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 ; MACRO CALLS
122 ;
123 .MCALL ISTAT$,STATE$,TRANS$
124
```



```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 .MACRO ERROR$ TEXT
65 ;
66 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
67 ; MESSAGE STRING.
68 ;
69 .IF DIF <TEXT><RO>
70 MOV TEXT,RO ; GET ADDRESS OF ERROR MESSAGE
71 .ENDC
72 CALL $CFERR ; PRINT ERROR MESSAGE
73 .ENDM ERROR$
74
75 .MACRO EPRINT TEXT
76 ;
77 ; PRINT TEXT ON ERROR LUN
78 ;
79 .IF DIF <TEXT><RO>
80 MOV TEXT,RO ; GET ADDRESS OF ASCIZ STRING TO PRINT
81 .ENDC
82 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
83 .ENDM EPRINT
84
85 ;****
86 ; MACRO CALLS
87 ;****
88 .MCALL ISTAT$,STATE$,TRANS$
89
90 ;****
91 ; LOCAL DATA
92 ;****
93
94 ;
95 ; DEFINE THE BUFFER BUFFER BLOCK
96 ;
97 .ASECT
98 =0
99 .BLKW 1 ; C.LNK
100 .BLKW 1 ; C.STS
101 C.CBNM: .BLKW 1 ; CCB NUMBER
102 C.CBSZ: .BLKW 1 ; CCB SIZE
103 C.RBNM: .BLKW 1 ; RDB NUMBER
104 C.RBSZ: .BLKW 1 ; RDB SIZE
105 C.SBNM: .BLKW 1 ; SDB NUMBER
106 C.SBSZ: .BLKW 1 ; SDB NUMBER
107 C.THSH: .BLKW 1 ; RDB THRESHOLD VALUE
108 C.LEN= . ; LENGTH OF THE BLOCK
109 .PSECT
110
111 NEXT: .BLKW 1 ; ADDRESS OF NEXT LOCATION IN TEMPLATE

```

LOCAL DATA

|     |        |        |     |     |         |                   |
|-----|--------|--------|-----|-----|---------|-------------------|
| 123 |        |        |     |     | .SBTTL  | LOCAL DATA        |
| 124 |        |        |     |     | .NLIST  | BEX               |
| 125 |        |        |     |     | .ASCII  | <11>/CHN\$DF/<11> |
| 126 | 000000 | 011    | 103 | 110 | CHNDF:  |                   |
| 127 |        | 000010 |     |     | CHNLN=. | -CHNDF            |
| 128 |        |        |     |     | .LIST   | BEX               |
| 129 |        |        |     |     | .EVEN   |                   |
| 130 |        |        |     |     |         |                   |

```

223 .SBTTL SETCSR - SET THE CSR ADDRESS
224 +
225 *** - SETCSR - SET THE CSR ADDRESS
226 :
227 INPUT:
228 .PNUMB = CSR ADDRESS
229 .PNUMH = CSR ADDRESS (HIGH ORDER)
230 TEMP = TEMPLATE ADDRESS
231 :
232 OUTPUT:
233 :
234 -
235
236 SETCSR:
237 000170 016702 177604 MOV TEMP,R2 ; GET START OF TEMPLATE
238 000174 016201 000010 MOV C.CSR(R2),R1 ; GET THE NEW CSR ADDRESS
239 000200 016700 000000G MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
240 000204 004767 000000G CALL $CNVO ; CONVERT NUMBER TO OCTAL
241 000210 010067 000000G MOV R0,$OBUF ; UPDATE THE POINTER INTO THE OUTPUT BUFFER
242 000214 000207
RETURN

```

UPDCUG - UPDATE USER GROUP DEF1 MACRO V05.03b Saturday 29-Jun-85 05:48 <sup>G 16</sup> Page 6-1  
Symbol table

|                  |                   |                   |                   |                  |
|------------------|-------------------|-------------------|-------------------|------------------|
| CS.FLG= ***** GX | C.LEN = 000014    | \$OBUF = ***** GX | \$PTOPT= ***** GX | .BUFF = ***** GX |
| CUGDF = 000000R  | C.NAM = 000004    | \$OFDB = ***** GX | \$PTR50= ***** GX | .COMMA= ***** GX |
| CUGLN = 000010   | F.NRBD= ***** GX  | \$PTNMB= ***** GX | \$UPCUG 000010RG  | .FLAGS= ***** GX |
| C.CUG = 000010   | \$BUFI = ***** GX | \$PTNUM= ***** GX | \$WDATA= ***** GX | .TEMP = ***** GX |
| C.FLG = 000012   | \$MVASC= ***** GX |                   |                   |                  |

. ABS. 000014 000 (RW,I,GBL,ARS,OVR)  
000170 001 (RW,I,LCL,REL,CON)

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 8838 Words ( 35 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:05.25

SY:UPDCUG.V2,[132,134]UPDCUG/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDCUG

TMPRDT - PARSE REMOTE DTE IN CE MACRO V05.03b Saturday 29-Jun-85 05:40<sup>1</sup> Page 8  
SETNAM - SET NAME

```
212 .SBTTL SETNAM - SET NAME
213 .SBTTL SETADR - SET ADDRESS
214 ;+
215 ;** SETNAM - SET NAME
216 ;** SETADR - SET ADDRESS
217 ;
218 ; INPUTS .TPARS VARIABLES
219 ;
220 ; OUTPUT VALUE SET IN TEMPLATE.
221 ; -
222
223 SETNAM: $GTR50 #R$DNAM,#2
224 SETADR: $GTR50 #R$DADR,#5
225
226 000200
227 000214
228 000001 .END
```

```

255 .SBTTL SETADD - SET UP REMOTE ADDRESS
256 :+
257 *** - SETADD - SETUP REMOTE ADDRESS
258 :
259 INPUT:
260 .PNUMB = THE REMOTE ADDRESS
261 TEMP = TEMPLATE ADDRESS
262 :
263 OUTPUT:
264 C.ADD = THE REMOTE NODE ADDRESS
265 :-
266
267
268 000324 SETADD:
269 000324 016700 000000G MOV .PNUMB,R0 ; PICK UP PARSED VALUE
270 000330 001410 BEQ 10$; OH COME ON, BE SERIOUS !
271 000332 020027 001777 CMP R0,#1023. ; COMPARE HIGH END OF RANGE
272 000336 101005 BHI 10$; IF HI, NG
273 000340 016701 177434 MOV TEMP,R1 ; GET THE REMOTE TEMPLATE
274 000344 050061 000010 BIS R0,C.ADD(R1) ; FOLD ADDRESS IN WITH AREA
275 000350 000402 BR 20$; ALL DONE
276 000352 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT TRANSITION
277 000356 000207 20$: RETURN
278
279
280 .END
000001

```

TMPROU - CFE PARSE ROUTING DEF1 MACRO V05.03b Saturday 29-Jun-85 05:40 H 3 Page 7-1  
Symbol table

|                                       |                               |                     |                   |                   |
|---------------------------------------|-------------------------------|---------------------|-------------------|-------------------|
| CS.ROU= ***** GX                      | EXIT 000074R                  | 002 SENBEA 000226R  | \$DIGIT= 000024   | \$SUBXP= 000010   |
| C.ACST 000022                         | LA = 000074                   | SETADD 000106R      | \$DNUMB= 000014   | \$TALOC= ***** GX |
| C.ADD 000004                          | MAXCST= ***** GX              | SETCST 000126R      | \$EQS = 000012    | \$TMROU 000002RG  |
| C.AHOP 000020                         | MAXHOP= ***** GX              | SETHOP 000146R      | \$ERROR= ***** GX | \$\$\$FLG= 177777 |
| C.CST 000006                          | MAXRTM= 177777                | SETNA 000206R       | \$EXIT = 000000   | \$\$\$KEY= 177777 |
| C.HOP 000010                          | RA = 000076                   | SETRTM 000166R      | \$FAIL = 177777   | \$\$\$STA= 000000 |
| C.LEN = 000026                        | ROUKTB 000000RG               | 003 TEMP 000000R    | \$GTNUM= ***** GX | .ERROR= ***** GX  |
| C.L2TM 000024                         | ROUSTB 000000RG               | 002 \$ALPHA= 000022 | \$LAMDA= 000000   | .FLAGS= ***** GX  |
| C.NA 000014                           | SEACST 000266R                | \$ANY = 000020      | \$NUMBR= 000002   | .TEMP = ***** GX  |
| C.NBEA 000016                         | SEAHOP 000246R                | \$BLANK= 000006     | \$RAD50= 000016   | .TPARS= ***** GX  |
| C.RTM 000012                          | SELV2T 000306R                | \$CFERR= ***** GX   | \$STRNG= 000004   |                   |
| . ABS. 000026 000 (RW,I,GBL,ABS,OVR)  |                               |                     |                   |                   |
|                                       | 000326 001 (RW,I,LCL,REL,CON) |                     |                   |                   |
| \$STATE 000104 002 (RW,D,LCL,REL,CON) |                               |                     |                   |                   |
| \$KIAB 000000 003 (RW,D,LCL,REL,CON)  |                               |                     |                   |                   |
| \$KSTR 000000 004 (RW,D,LCL,REL,CON)  |                               |                     |                   |                   |
| Errors detected: 0                    |                               |                     |                   |                   |

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10750 Words ( 42 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:28.46  
SY: TMPROU.V2,[132,134]TMPROU/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPROU

TMPSE - CFE PARSE REMOTE CHARA MACRO V05.03b Saturday 29-Jun-85 05:41 Page 7-1  
 SETARE - SET UP REMOTE AREA

|     |        |        |        |        |             |                                        |
|-----|--------|--------|--------|--------|-------------|----------------------------------------|
| 316 | 000236 | 010011 |        | MOV    | R0,(R1)     | ; SET AREA, INITIALIZE ADDRESS TO ZERO |
| 317 | 000240 | 000402 |        | BR     | 20\$        | ; ALL DONE                             |
| 318 | 000242 |        |        |        |             |                                        |
| 319 | 000242 | 062716 | 000002 | REJ:   |             |                                        |
| 320 | 000246 | 000207 |        | 10\$:  | ADD #2,(SP) | ; ELSE, REJECT TRANSITION              |
| 321 |        |        |        | 20\$:  | RETURN      |                                        |
| 322 |        |        |        | .DSABL | LSB         |                                        |



```

112
113
114
115
116
117
118 000000
119
120
121
122
123
124 000002 004 103 106
125 000034 004 103 106
126 000066 004 103 106
127 000120 004 103 106
128 000147 004 103 106
129 000212 004 103 106
130 000247 004 103 106
131 000302 004 103 106
132 000337 004 103 106
133
134

```

```

.MCALL ISTAT$,STAT$,TRANS$

:****
: LOCAL DATA
:****
FLAG: .BLKW 1 ; TEMPLATE FLAGS WORD
:
: ERROR STRINGS
:
EX$SEV = 4
.NLIST BEX
.ERRNAM: .ASCIZ <EX$SEV>/CFE -- Error in DDM name/
.ERRDLC: .ASCIZ <EX$SEV>/CFE -- Error in DLC name/
.ERRLLC: .ASCIZ <EX$SEV>/CFE -- Error in LLC name/
.ERRFLG: .ASCIZ <EX$SEV>/CFE -- Error in flags/
.ERRCNT: .ASCIZ <EX$SEV>/CFE -- Error in controller number/
.ERRUNT: .ASCIZ <EX$SEV>/CFE -- Error in unit number/
.ERRTYP: .ASCIZ <EX$SEV>/CFE -- Error in line type/
.ERRHTM: .ASCIZ <EX$SEV>/CFE -- Error in hello timer/
.ERRCTT: .ASCIZ <EX$SEV>/CFE -- Error in counter timer/
.EVEN
.LIST BEX

```

TMPSLT      CREATED BY MACRO ON 29-JUN-85 AT 05:43      PAGE 2      H 6  
 SYMBOL CROSS REFERENCE      CREF 04.00

| SYMBOL    | VALUE      | REFERENCES                                             |
|-----------|------------|--------------------------------------------------------|
| \$GTNMB   | = ***** GX | 9-334      10-350                                      |
| \$GTNUM   | = ***** GX | 11-364      12-381      12-383      12-385             |
| \$GTR50   | = ***** GX | 7-301                                                  |
| \$LAMDA   | = 000000   | #5-140                                                 |
| \$NUMBR   | = 000002   | #5-140                                                 |
| \$RAD50   | = 000016   | #5-140                                                 |
| \$RONLY   | = *****    | 5-140      5-140                                       |
| \$STAN    | = ***** GX | *6-265                                                 |
| \$STRNG   | = 000004   | #5-140                                                 |
| \$SUBXP   | = 000010   | #5-140                                                 |
| \$TALOC   | = ***** GX | 6-268                                                  |
| \$TMSLT   | 000376 RG  | #6-263                                                 |
| \$\$\$FLG | = 177777   | #5-140                                                 |
| \$\$\$KEY | = 177777   | #5-140                                                 |
| .ERROR    | = ***** GX | 6-278                                                  |
| .FLAGS    | = ***** GX | *6-271      11-364      12-381      12-383      12-385 |
| .TEMP     | = ***** GX | *6-269      8-317      13-404                          |
| .TPARS    | = ***** GX | 6-276                                                  |

```

301 .SBTTL SETHTM - SETUP STATION HELLO TIMER
302 ;+
303 ;*** - SETHTM - SETUP STATION HELLO TIMER
304 ;
305 ;THIS ACTION ROUTINE SETS UP THE STATION TIMER
306 ;
307 ;INPUT:
308 ;.PNUMB = STATION HELLO TIMER
309 ;.PNUMH = STATION HELLO TIMER (HIGH ORDER)
310 ;
311 ;OUTPUT:
312 ;C.HTM = STATION HELLO TIMER
313 ;
314 ;-
315
316 000322 SETHTM: MOV TEMP,RO ; GET TEMPLATE COST
317 000322 016700 177452 BIS #ST$HTM,C.STS(RO) ; REMEMBER OPTION
318 000326 052760 000000G 000000G MOV .PNUMB,C.HTM(RO) ; STORE THE STATION HELLO TIMER IN TEMPLATE
319 000334 016760 000000G 000012 TST .PNUMH ; IS IT A WORD VALUE ?
320 000342 005767 000000G BEQ 10$; IF EQ, YES - OKAY
321 000346 001402 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
322 000350 062716 000002 10$: RETURN
323 000354 000207
324
325 000001 .END

```

```

271 .SBTTL SETNAM - SET DLM NAME
272 .SBTTL SETPSZ - SET PACKET SIZE
273 .SBTTL SETWSZ - SET WINDOW SIZE
274 .SBTTL SETFLG - SET FLAGS
275 .SBTTL SETRCL - SET MAXIMUM RECALLS
276 .SBTTL SETTLM - SET RECALL TIMER
277 .SBTTL SETDTE - SET DTE ADDRESS
278 .SBTTL SETOWN - SET OWNER OF SVC
279
280 ;+
281 ; ** SETNAM - SET DLM NAME
282 ; ** SETPSZ - SET PACKET SIZE
283 ; ** SETWSZ - SET WINDOW SIZE
284 ; ** SETFLG - SET FLAGS
285 ; ** SETRCL - SET MAXIMUM RECALLS
286 ; ** SETTLM - SET RECALL TIMER
287 ; ** SETDTE - SET DTE ADDRESS
288 ; ** SETOWN - SET OWNER OF SVC
289
290 ; INPUTS .TPARS VARIABLES.
291
292 ; OUTPUT VALUE SET IN TEMPLATE.
293 ; -
294
295 000454 SETNM1: $GTNUM $$SVNAM ; NAME (XX OF DLM-XX.YY)
296
297 000474 SETNM2: $GTNUM $$SVNAM+2 ; NAME (YY OF DLM-XX.YY)
298
299 000514 SETPSZ: $GTNUM $$SVPSZ,,,SV$PSZ ; PACKET SIZE
300
301 000542 SETWSZ: $GTNUM $$SVWSZ,,,SV$WSZ ; WINDOW SIZE
302
303 000570 SETFLG: $GTNUM $$SVFLG ; FLAGS
304
305 000610 SETRCL: $GTNUM $$SVRCL ; MAXIMUM RECALLS
306
307 000630 SETTLM: $GTNUM $$SVTLM ; RECALL TIMER
308
309 000650 SETDTE: $GTR50 $$VDTE,#5,SVDTE ; RAD50 OF DTE ADDRESS
310
311 000672 SETOWN: $GTR50 $$SVOWN,#1 ; RAD50 OWNER OF SVC
312
313 .END

```

000001

```

228 .SBTTL SETCW1 - SET THE CHARACTERISTICS WORD 1
229
230 ;+
231 ; *** - SETCW1 - SET THE CONTROLLER CHARACTERISTICS WORD 1
232 ; THIS ACTION ROUTINE CHECKS THE CHARACTERISTICS WORD 1 TO MAKE SURE IT IS VALID
233 ; AND STORES IT IN THE TEMPLATE.
234
235 ; INPUT:
236 ; .PNUMB = CHARACTERISTICS WORD 1
237 ; .PNUMH = CHARACTERISTICS WORD 1 (HIGH ORDER)
238 ; .TEMP = TEMPLATE ADDRESS
239
240 ; OUTPUT:
241 ; THE CHARACTERISTICS WORD 1 IS STORED IN C.CW1 OF THE TEMPLATE.
242
243 ; -
244
245 000144 SETCW1: MOV .TEMP,R0 ; GET START OF TEMPLATE
246 000144 016700 MOV .PNUMB,C.CW1(R0) ; STORE THE CHARACTERISTICS WORD 1 IN THE TEMPLATE
247 000150 016760 TST .PNUMH ; IS IT A WORD VALUE ?
248 000156 005767 BEQ 10$; IF EQ, YES - OKAY
249 000162 001402 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
250 000164 062716
251 000170 000207 10$: RETURN

```

```

126 .SBTTL LOCAL DATA
127
128 ; DEFINE THE LEVEL 2 PARAMETER BLOCK
129 ; ** DO NOT CHANGE THESE OFFSETS WITHOUT REDEFINING THE X$2xxx CONSTANTS !
130
131 ;
132 .ASECT
133 .=0
134 000000 000000
135 .BLKW 1 ; C.LNK
136 .BLKW 1 ; C.STS
137 C.LIN: .BLKW 1 ; LINE ID: RAD50 PROCESS NAME
138 .BLKW 2 ; CONTROLLER, UNIT NUMBERS
139 C.WSZ: .BLKW 1 ; WINDOW SIZE (0-7)
140 C.BSZ: .BLKW 1 ; BLOCK SIZE
141 C.RTC: .BLKW 1 ; RETRANSMIT COUNT
142 C.RTT: .BLKW 1 ; RETRANSMIT TIMER VALUE
143 C.HBT: .BLKW 1 ; HOLDBACK TIMER VALUE
144 C.LEN=.
145 .PSECT
146 .NLIST BEX
147
148 ; SYNTAX ERROR DETAIL STRINGS
149 ;
150 EX$SEV = 4
151 000000 004 103 106 ERRLIN: .ASCIZ <EX$SEV>/CFE -- Invalid line id/
152 000030 004 103 106 ERRWSZ: .ASCIZ <EX$SEV>/CFE -- Error in window size/
153 000065 004 103 106 ERRBSZ: .ASCIZ <EX$SEV>/CFE -- Invalid block size/
154 000120 004 103 106 ERRRTC: .ASCIZ <EX$SEV>/CFE -- Invalid retransmit count /
155 000162 004 103 106 ERRRTT: .ASCIZ <EX$SEV>/CFE -- Invalid retransmit timer value/
156 000231 004 103 106 ERRHBT: .ASCIZ <EX$SEV>/CFE -- Invalid holdback timer value/
157 .EVEN
158 .LIST BEX

```

```

114 MOV LEN,R1
115 IF NB <OPT>
116 BIS OPT,@.FLAGS
117 .ENDC
118 JMP $GTR50
119 .ENDM
120
121 ; MACRO CALLS
122 ;
123 .MCALL ISTAT$,STAT$,TRAN$
124

```

```

126 .SBTTL LOCAL DATA
127
128 ;
129 ; DEFINE THE X.25 LEVEL 3 PARAMETER BLOCK
130 ;
131 000000 .ASECT
132 000000 000000 .=0
133 000000 .BLKW 1 ; C.LNK
134 000002 .BLKW 1 ; C.STS
135 000004 C.DFP: .BLKW 1 ; DEFAULT BLOCK SIZE (21.-1029.)
136 000006 C.MXP: .BLKW 1 ; MAXIMUM BLOCK SIZE (21.-1029.)
137 000010 C.DW: .BLKB 1 ; DEFAULT WINDOW SIZE (1.-127.)
138 000011 C.MW: .BLKB 1 ; MAXIMUM WINDOW SIZE (1.-127.)
139 000012 C.CT: .BLKB 1 ; CALL TIMER VALUE (0.-255.)
140 000013 C.KT: .BLKB 1 ; CLEAR TIMER VALUE (0.-255.)
141 000014 C.RT: .BLKB 1 ; RESET TIMER VALUE (0.-255.)
142 000015 C.ST: .BLKB 1 ; RESTART TIMER VALUE (0.-255.)
143 000016 C.KM: .BLKB 1 ; MAXIMUM CLEAR VALUE (0.-255.)
144 000017 C.RM: .BLKB 1 ; MAXIMUM RESETS VALUE (0.-255.)
145 000020 C.SM: .BLKB 1 ; MAXIMUM RESTARTS VALUE (0.-255.)
146 000021 C.LEN=. ; LENGTH OF THE BLOCK
147 000000 .PSECT
148
149 ; SYNTAX ERROR DETAIL STRINGS
150 ;
151 .NLIST BEX
152 000004 000004 103 106 ERRDFP: .ASCIZ <EX$SEV>/CFE -- Invalid default block size/
153 000000 004 103 106 ERRMXP: .ASCIZ <EX$SEV>/CFE -- Invalid maximum block size/
154 000043 004 103 106 ERRDW: .ASCIZ <EX$SEV>/CFE -- Invalid default window size/
155 000106 004 103 106 ERRMW: .ASCIZ <EX$SEV>/CFE -- Invalid maximum window size/
156 000152 004 103 106 ERRCT: .ASCIZ <EX$SEV>/CFE -- Invalid call timer value/
157 000216 004 103 106 ERRKT: .ASCIZ <EX$SEV>/CFE -- Invalid clear timer value/
158 000257 004 103 106 ERRRT: .ASCIZ <EX$SEV>/CFE -- Invalid reset timer value/
159 000321 004 103 106 ERRST: .ASCIZ <EX$SEV>/CFE -- Invalid restart timer value/
160 000363 004 103 106 ERRKM: .ASCIZ <EX$SEV>/CFE -- Invalid maximum clear value/
161 000427 004 103 106 ERRRM: .ASCIZ <EX$SEV>/CFE -- Invalid maximum resets value/
162 000473 004 103 106 ERRSM: .ASCIZ <EX$SEV>/CFE -- Invalid maximum restarts value/
163 000540 004 103 106
164 .EVEN
165 .LIST BEX

```



```

113 ;****
114 ; INITIALIZE TABLES
115 ;****
116
117 ISTAT$ BUFSTB,BUFKTB
118
119 ; PROCESS 'BUF$DF'
120 ;
121 STATES$
122 TRANS$ $NUMBR,,SETNUM
123
124 STATES$
125 TRANS$ <','>,, $MOVPT
126
127 STATES$
128 TRANS$ $NUMBR,,SETNUM
129
130 STATES$
131 TRANS$ <','>,, $MOVPT
132
133 STATES$
134 TRANS$ $NUMBR,,SETNUM
135
136 STATES$
137 TRANS$ <','>,, $MOVPT
138
139 STATES$
140 TRANS$ $NUMBR,,SETNUM
141
142 STATES$
143 TRANS$ <','>,, $MOVPT
144
145 STATES$
146 TRANS$ $NUMBR,,SETNUM
147
148 STATES$
149 TRANS$ <','>,, $MOVPT
150
151 STATES$
152 TRANS$ $NUMBR,,SETNUM
153
154 STATES$
155 TRANS$ <','>,, $MOVPT
156
157 STATES$
158 TRANS$ $NUMBR,,SETNUM
159
160 STATES$ EXIT
161 TRANS$ $EOS,$EXIT,$CLNUP
162 TRANS$ <','>,, $MOVPT
163 STATES$
164 TRANS$ $NUMBR,, $MOVPT
165 STATES$
166 TRANS$ $LAMDA,EXIT
167
168 ;
169

```

```

132 .SBTTL $UPCHN - UPDATE CHANNEL DEFINITION
133 ;+
134 *** - $UPCHN - UPDATE CHANNEL DEFINITION
135 ;
136 INPUT:
137 R5 - TEMPLATE ADDRESS
138 ;
139 OUTPUT:
140 C-BIT = SUCCESS/FAILURE
141 ;
142 -
143 ;
144 000010 $UPCHN::
145 000010 012767 000000G 000000C MOV #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
146 000016 010567 000000G MOV R5,TEMP ; SAVE TEMPLATE ADDRESS
147 000022 005725 TST (R5)+ ; POINT TO C.STS
148 000024 010567 000000G MOV R5,FLAGS ; SAVE ADDRESS OF FLAGS WORD
149 000030 012767 177777 000000G MOV #-1,COMMA ; SUPPRESS INITIAL COMMA
150 000036 012700 000000G MOV #.BUFF,R0 ; POINT AT START OF OUTPUT BUFFER
151 000042 012701 000000' MOV #CHNDF,R1 ; POINT AT START OF CHANNEL DEFINITION
152 000046 012702 000010 MOV #CHNLN,R2 ; GET LENGTH OF CHN DEFINITION
153 000052 004767 000000G CALL $MVASC ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
154 000056 010067 000000G MOV R0,$OBUF ; SAVE CURRENT POSITION IN OUTPUT BUFFER
155 000062 $PTNUM #C$HLO ; SET CHANNEL LOW
156 000074 $PTNUM #C$HHI, #C$SHI ; SET CHANNEL HIGH
157 000120 016701 000000G MOV $OBUF,R1 ; CALCULATE LENGTH OF BUFFER
158 000124 162701 000000G SUB #.BUFF,R1 ;
159 000130 010167 000000C MOV R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
160 000134 004767 000000G CALL $WDATA ; WRITE THE RECORD TO THE OUTPUT FILE
161 000140 012767 000000G 000000C MOV #$BUF1,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE 'END$DF' LINE
162 000146 000207 RETURN
163
164 000001 .END

```

244  
 245  
 246  
 247  
 248  
 249  
 250  
 251  
 252  
 253  
 254  
 255  
 256  
 257  
 258  
 259  
 260  
 261  
 262  
 263

000216  
 000216 016702 177556  
 000222 016201 000012  
 000226 016700 000000G  
 000232 004767 000000G  
 000236 010067 000000G  
 000242 000207

```

.SBTTL SETPRI - SET THE PRIORITY
*** - SETPRI - SET THE CONTROLLER PRIORITY
INPUT:
.PNUMB = PRIORITY
.PNUMH = PRIORITY (HIGH ORDER)
TEMP = TEMPLATE ADDRESS
OUTPUT:
-
SETPRI:
MOV TEMP,R2 ; GET START OF TEMPLATE
MOV C.PRI(R2),R1 ; GET THE PRIORITY
MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
CALL $CNVO ; CONVERT NUMBER TO OCTAL
MOV R0,$OBUF ; UPDATE THE POINTER INTO THE OUTPUT BUFFER
RETURN

```

UPDCUG      CREATED BY    MACRO    ON 29-JUN-85 AT 05:48      PAGE 1      H 16

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE             | REFERENCES                 |
|---------|-------------------|----------------------------|
| CS.FLG  | =    *****    GX  | 6-178                      |
| CUGDF   | =    000000    R  | #5-146    5-147    6-171   |
| CUGLN   | =    000010       | #5-147    6-172            |
| C.CUG   | =    000010       | #5-141    6-176            |
| C.FLG   | =    000012       | #5-142    6-178            |
| C.LEN   | =    000014       | #5-143                     |
| C.NAM   | =    000004       | #5-140    6-175            |
| F.NRBD  | =    *****    GX  | *6-165    *6-181    *6-183 |
| \$BUFI  | =    *****    GX  | 6-183                      |
| \$MVASC | =    *****    GX  | 6-173                      |
| \$OBUF  | =    *****    GX  | *6-174    *6-177    6-179  |
| \$OFDB  | =    *****    GX  | *6-165    *6-181    *6-183 |
| \$PTNMB | =    *****    GX  | 6-178                      |
| \$PTNUM | =    *****    GX  | 6-176                      |
| \$PTOPT | =    *****    GX  | 6-178                      |
| \$PTR50 | =    *****    GX  | 6-175                      |
| \$UPCUG | =    000010    RG | #6-164                     |
| \$WDATA | =    *****    GX  | 6-182                      |
| .BUFF   | =    *****    GX  | 6-165    6-170    6-180    |
| .COMMA  | =    *****    GX  | *6-169                     |
| .FLAGS  | =    *****    GX  | *6-168                     |
| .TEMP   | =    *****    GX  | *6-166                     |

1 1  
 TMPRDT - PARSE REMOTE DTE IN CE MACRO V05.03b Saturday 29-Jun-85 05:40 Page 8-1  
 Symbol table

|                   |                       |                   |                   |                   |
|-------------------|-----------------------|-------------------|-------------------|-------------------|
| CS.RDT= ***** GX  | R\$DLEN= ***** GX     | \$BYTMN= ***** GX | \$FAIL = 177777   | \$TMRDT 000066RG  |
| DTE 000040R       | 002 R\$DNAM= ***** GX | \$CFERR= ***** GX | \$GTR50= ***** GX | \$\$\$FLG= 177777 |
| DTE1 000042R      | 002 R\$LENX= ***** GX | \$CLERR= ***** GX | \$LAMD= 000000    | \$\$\$KEY= 177777 |
| ERRADR 000031R    | SETADR 000214R        | \$DIGIT= 000024   | \$NUMBR= 000002   | \$\$\$STA= 000000 |
| ERRNAM 000000R    | SETNAM 000200R        | \$DNUMB= 000014   | \$RAD50= 000016   | .ERROR= ***** GX  |
| EX\$SEV= 000004   | \$ALPHA= 000022       | \$EOS = 000012    | \$STRNG= 000004   | .FLAGS= ***** GX  |
| RDTKTB 000000RG   | 003 \$ANY = 000020    | \$ERROR= ***** GX | \$SUBXP= 000010   | .TEMP = ***** GX  |
| RDTSTB 000000RG   | 002 \$BLANK= 000006   | \$EXIT = 000000   | \$TALOC= ***** GX | .TPARS= ***** GX  |
| R\$DADR= ***** GX |                       |                   |                   |                   |

. ABS. 000000 000 (RW,I,GBL,ABS,OVR)  
 000230 001 (RW,I,LCL,REL,CON)  
 \$STATE 000052 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000000 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000000 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10992 Words ( 43 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:17.22  
 SY:TMPRDT.V2,[132,134]TMPRDT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPRDT

TMPREM - CFE PARSE REMOTE DEFIN MACRO V05.03b Saturday 29-Jun-85 05:40 Page 10-1  
Symbol table

|                  |                     |                       |                   |                   |
|------------------|---------------------|-----------------------|-------------------|-------------------|
| ADD2 000024R     | 002 REMKTB 000000RG | 003 \$BYTMN= ***** GX | \$LSNAD= ***** GX | \$TMREM 000002RG  |
| CS.REM= ***** GX | REMSTB 000000RG     | 002 \$CAT5 = ***** GX | \$LSTMP= ***** GX | \$T1ALC= ***** GX |
| C.ADD 000010     | R.LENX= ***** GX    | \$DIGIT= 000024       | \$NTADD= ***** GX | \$\$\$FLG= 177777 |
| C.LEN = 000012   | SETADD 000324R      | \$DNUMB= 000014       | \$NUMBR= 000002   | \$\$\$KEY= 177777 |
| C.NAM 000004     | SETARE 000262R      | \$EOS = 000012        | \$QINSN= ***** GX | \$\$\$STA= 000000 |
| C.STS = ***** GX | SETNAM 000202R      | \$ERROR= ***** GX     | \$RAD50= 000016   | .PNUMB= ***** GX  |
| EXIT 000030R     | TEMP 000000R        | \$EXIT = 000000       | \$STRNG= 000004   | .PSTCN= ***** GX  |
| FINREM 000114R   | \$ALPHA= 000022     | \$FAIL = 177777       | \$SUBXP= 000010   | .PSTPT= ***** GX  |
| LA = 000074      | \$ANY = 000020      | \$LAMDA= 000000       | \$TMLST= ***** GX | .TPARS= ***** GX  |
| RA = 000076      | \$BLANK= 000006     |                       |                   |                   |

. ABS. 000012 000 (RW,I,GBL,ABS,OVR)  
000360 001 (RW,I,LCL,REL,CON)  
\$STATE 000034 002 (RW,D,LCL,REL,CON)  
\$KTAB 000000 003 (RW,D,LCL,REL,CON)  
\$KSTR 000000 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10406 Words ( 41 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:17.19  
SY:TMPREM.V2,[132,134]TMPREM/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPREM

| SYMBOL    | VALUE     | REFERENCES |        |        |       |       |       |       |       |
|-----------|-----------|------------|--------|--------|-------|-------|-------|-------|-------|
| CS.ROU    | = *****   | GX         | 6-228  |        |       |       |       |       |       |
| C.ACST    | 000022    |            | #4-136 | 7-268  |       |       |       |       |       |
| C.ADD     | 000004    |            | #4-129 | 7-247  |       |       |       |       |       |
| C.AHOP    | 000020    |            | #4-135 | 7-265  |       |       |       |       |       |
| C.CST     | 000006    |            | #4-130 | 7-250  |       |       |       |       |       |
| C.HOP     | 000010    |            | #4-131 | 7-253  |       |       |       |       |       |
| C.LEN     | = 000026  |            | #4-138 | 6-223  |       |       |       |       |       |
| C.L2TM    | 000024    |            | #4-137 | 7-271  |       |       |       |       |       |
| C.NA      | 000014    |            | #4-133 | 7-259  |       |       |       |       |       |
| C.NBEA    | 000016    |            | #4-134 | 7-262  |       |       |       |       |       |
| C.RTM     | 000012    |            | #4-132 | 7-256  |       |       |       |       |       |
| LA        | = 000074  |            | #4-118 |        |       |       |       |       |       |
| MAXCST    | = *****   | GX         | 7-250  |        |       |       |       |       |       |
| MAXHOP    | = *****   | GX         | 7-253  |        |       |       |       |       |       |
| MAXRTM    | = 177777  |            | #4-120 | 7-256  |       |       |       |       |       |
| RA        | = 000076  |            | #4-119 |        |       |       |       |       |       |
| ROUKTB    | 000000    | RG         | #5-147 | 6-231  |       |       |       |       |       |
| ROUSTB    | 000000    | RG         | #5-147 | 6-232  |       |       |       |       |       |
| SEACST    | 000266    | R          | #7-267 |        |       |       |       |       |       |
| SEAHOP    | 000246    | R          | #7-264 |        |       |       |       |       |       |
| SELV2T    | 000306    | R          | #7-270 |        |       |       |       |       |       |
| SENBEA    | 000226    | R          | #7-261 |        |       |       |       |       |       |
| SETADD    | 000106    | R          | #7-246 |        |       |       |       |       |       |
| SETCST    | 000126    | R          | #7-249 |        |       |       |       |       |       |
| SETHOP    | 000146    | R          | #7-252 |        |       |       |       |       |       |
| SETNA     | 000206    | R          | #7-258 |        |       |       |       |       |       |
| SETRTM    | 000166    | R          | #7-255 |        |       |       |       |       |       |
| TEMP      | 000000    | R          | #4-121 |        |       |       |       |       |       |
| \$ALPHA   | = 000022  |            | #5-147 |        |       |       |       |       |       |
| \$ANY     | = 000020  |            | #5-147 |        |       |       |       |       |       |
| \$BLANK   | = 000006  |            | #5-147 |        |       |       |       |       |       |
| \$CFERR   | = *****   | GX         | 6-237  |        |       |       |       |       |       |
| \$DIGIT   | = 000024  |            | #5-147 |        |       |       |       |       |       |
| \$DNUMB   | = 000014  |            | #5-147 |        |       |       |       |       |       |
| \$EOS     | = 000012  |            | #5-147 |        |       |       |       |       |       |
| \$ERROR   | = *****   | GX         | *6-238 | *6-239 |       |       |       |       |       |
| \$EXIT    | = 000000  |            | #5-147 |        |       |       |       |       |       |
| \$FAIL    | = 177777  |            | #5-147 |        |       |       |       |       |       |
| \$GPRM    | = *****   |            | 5-147  |        |       |       |       |       |       |
| \$GTNUM   | = *****   | GX         | 7-247  | 7-250  | 7-253 | 7-256 | 7-259 | 7-262 | 7-265 |
| \$LAMDA   | = 000000  |            | #5-147 |        |       |       |       |       | 7-268 |
| \$NUMBR   | = 000002  |            | #5-147 |        |       |       |       |       | 7-271 |
| \$RAD50   | = 000016  |            | #5-147 |        |       |       |       |       |       |
| \$RONLY   | = *****   |            | 5-147  | 5-147  | 5-147 |       |       |       |       |
| \$STRNG   | = 000004  |            | #5-147 |        |       |       |       |       |       |
| \$SUBXP   | = 000010  |            | #5-147 |        |       |       |       |       |       |
| \$TALQC   | = *****   | GX         | 6-224  |        |       |       |       |       |       |
| \$TMROU   | 000002    | RG         | #6-221 |        |       |       |       |       |       |
| \$\$\$FLG | = 177777  |            | #5-147 |        |       |       |       |       |       |
| \$\$\$KEY | = 177777  |            | #5-147 |        |       |       |       |       |       |
| .ERROR    | = *****   | GX         | *6-229 | 6-235  |       |       |       |       |       |
| .FLAGS    | = ** :*** | GX         | *6-227 |        |       |       |       |       |       |

```

324 .SBTTL STCIR - STORE SERVICE CIRCUIT
325
326 *** STCIR - STORE SERVICE CIRCUIT
327
328 INPUT: .PSTPT = ADDRESS OF STRING
329 .PSTCN = STRING LENGTH
330 .TEMP = TEMPLATE ADDRESS
331
332 OUTPUT:
333
334 $$$CIR = SERVICE CIRCUIT (RAD50)
335
336 000250 026727 000000G 000030 STCIR: CMP .PSTCN,#<$$$DEV-$$$CIR>*3 ; LENGTH OK ?
337 000256 101371 BHI REJ ; IF HI, NO - REJECT TRANSITION
338 000260 SAVRG R3 ; ELSE SAVE REGISTER
339 000262 012703 000022 MOV #$$$CIR,R3 ; GET OFFSET OF CIRCUIT IN TEMPLATE
340 000266 066703 000000G ADD .TEMP,R3 ; POINT INTO TEMPLATE
341 000272 016700 000000G MOV .PSTPT,R0 ; POINT TO PARSED STRING
342 000276 111446 MOVB (R4),-(SP) ; SAVE NEXT CHARACTER
343 000300 112714 000077 MOVB #<'?'>,(R4) ; INSERT STOP CHARACTER
344 000304 004767 000000G 10$: CALL .CAT5E ; CONVERT TO RAD50 (EXTENDED WITH '-')
345 000310 010123 MOV R1,(R3)+ ; STORE CONVERTED VALUE
346 000312 103374 BCC 10$; IF CC, MORE
347 000314 112614 MOVB (SP)+,(R4) ; RESTORE TRAILING CHARACTER
348 000316 RESRG R3
349 000320 000207 RETURN
350

```



```

136 ;****
137 ; INITIALIZE TABLE GENERATION
138 ;****
139
140 000376 ISTAT$ SLTSTB,SLTKTB
141
142 000376 STATES$
143 000376 TRANS$ $LAMDA,, $CLERR,ERRNAM,,.ERROR ; DDM
144
145 000376 STATES$
146 000376 TRANS$ $RAD50,,SETNAM
147
148 000376 STATES$
149 000376 TRANS$ <','>,, $CLERR,ERRDLC,,.ERROR ; DLC
150
151 000376 STATES$
152 000376 TRANS$ $RAD50
153
154 000376 STATES$
155 000376 TRANS$ <','>,, $CLERR,ERRLLC,,.ERROR ; LLC
156
157 000376 STATES$
158 000376 TRANS$ $RAD50
159
160 000376 STATES$
161 000376 TRANS$ <','>,, $CLERR,ERRFLG,,.ERROR ; FLAGS
162
163 000376 STATES$
164 000376 TRANS$!LFS
165
166 000376 STATES$
167 000376 TRANS$!BITS
168
169 000376 STATES$
170 000376 TRANS$ $LAMDA,, $CLERR,ERRCNT,,.ERROR ; CONTROLLER NUMBER
171
172 000376 STATES$
173 000376 TRANS$ $NUMBR,,SETCNT
174
175 000376 STATES$
176 000376 TRANS$ <','>,, $CLERR,ERRUNT,,.ERROR ; UNIT NUMBER
177
178 000376 STATES$
179 000376 TRANS$ $NUMBR,,SETUNT
180
181 000376 STATES$
182 000376 TRANS$ $EOS,EXIT
183 000376 TRANS$ <','>,, $CLERR,ERRTYP,,.ERROR ; OPTIONAL TYPE
184
185 000376 STATES$
186 000376 TRANS$ 'MASTER',CTT,SETMST
187 000376 TRANS$ 'SLAVE',CTT,SETSLV
188 000376 TRANS$ $LAMDA
189
190 000376 STATES$ CTT
191 000376 TRANS$ $EOS,EXIT
192 000376 TRANS$ <','>,, $CLERR,ERRCTT,,.ERROR ; OPTIONAL COUNTER TIMER

```

TMPSLT      CREATED BY    MACRO    ON 29-JUN-85 AT 05:43      PAGE 3      I 6  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-140     | #5-183 | #5-187 | #5-188 | #5-192 | #5-196 | #5-200 | #5-204 | #5-208 | #5-211 |
|            | #5-215     | #5-228 | #5-235 | #5-236 | #5-237 | #5-238 | #5-239 | #5-240 | #5-241 |        |
| ERROR\$    | #4-64      | 6-280  |        |        |        |        |        |        |        |        |
| ISTAT\$    | #4-113     | 5-140  |        |        |        |        |        |        |        |        |
| MTRANS     | #5-140     |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-59      |        |        |        |        |        |        |        |        |        |
| STAT\$     | #4-113     | 5-142  | #5-145 | #5-148 | #5-151 | #5-154 | #5-157 | #5-160 | #5-163 | #5-166 |
|            | #5-169     | #5-172 | #5-175 | #5-178 | #5-181 | #5-185 | #5-190 | #5-194 | #5-198 | #5-202 |
|            | #5-206     | #5-209 | #5-213 | #5-216 | #5-220 | #5-223 | #5-226 | #5-230 | #5-233 | #5-245 |
|            | #5-248     |        |        |        |        |        |        |        |        |        |
| TRANS      | #4-113     | #5-143 | #5-146 | #5-149 | #5-152 | #5-155 | #5-158 | #5-161 | #5-164 | #5-167 |
|            | #5-170     | #5-173 | #5-176 | #5-179 | #5-182 | #5-183 | #5-186 | #5-187 | #5-188 | #5-191 |
|            | #5-192     | #5-195 | #5-196 | #5-199 | #5-200 | #5-203 | #5-204 | #5-207 | #5-208 | #5-210 |
|            | #5-211     | #5-214 | #5-215 | #5-217 | #5-221 | #5-224 | #5-227 | #5-228 | #5-231 | #5-234 |
|            | #5-235     | #5-236 | #5-237 | #5-238 | #5-239 | #5-240 | #5-241 | #5-246 |        |        |
| \$GTNMB    | #4-88      | 9-334  | 10-350 |        |        |        |        |        |        |        |
| \$GTNUM    | #4-78      | 11-364 | 12-381 | 12-383 | 12-385 |        |        |        |        |        |
| \$GTR50    | #4-100     | 7-301  |        |        |        |        |        |        |        |        |

I 7

TMPSTA - CFE PARSE STATION DEFI MACRO V05.03b Saturday 29-Jun-85 05:43 Page 10-1  
Symbol table

|                  |     |                    |                     |                       |                    |     |
|------------------|-----|--------------------|---------------------|-----------------------|--------------------|-----|
| APRNUM 000052R   | 002 | EX\$SEV= 000004    | SF.SER= ***** GX    | \$DIGIT= 000024       | \$STRNG= 000004    |     |
| BITS 000112R     | 002 | FLAG 000002R       | SF.SVC= ***** GX    | \$DNUMB= 000014       | \$SUBXP= 000010    |     |
| CS.STA= ***** GX |     | HTMR 000066R       | 002 SKIP 000102R    | \$EOS = 000012        | \$TALOC= ***** GX  |     |
| C.ADD 000005     |     | MXCSTL= ***** GX   | STACST 000036R      | 002 \$ERROR= ***** GX | \$TMSTA 000036RG   |     |
| C.APR 000010     |     | SETADD 000126R     | STAKTB 000000RG     | 003 \$EXIT = 000000   | \$\$\$FLG= 177777  |     |
| C.CST 000007     |     | SETAPR 000266R     | STASTB 000000RG     | 002 \$FAIL = 177777   | \$\$\$KEY= 000004  |     |
| C.FLG 000006     |     | SETCST 000216R     | ST\$HTM= ***** GX   | \$LAMDA= 000000       | \$\$\$STA= 000102R | 002 |
| C.HTM 000012     |     | SETHTM 000322R     | TEMP 000000R        | \$MXSTA= ***** GX     | \$\$\$TMP= 000017R | 004 |
| C.LEN = 000016   |     | SFLAG 000020R      | 002 \$ALPHA= 000022 | \$NUMBR= 000002       | .PNUMB= ***** GX   |     |
| C.STA 000004     |     | SFS 000074R        | 002 \$ANY = 000020  | \$RAD50= 000016       | .PNUMH= ***** GX   |     |
| C.STS = ***** GX |     | SF.ENA= ***** GX   | \$BLANK= 000006     | \$STAN = ***** GX     | .TPARS= ***** GX   |     |
| ERRSTA 000004R   |     | SF.MFL= ***** GX   | \$CFERR= ***** GX   |                       |                    |     |
|                  |     |                    |                     |                       |                    |     |
| . ABS. 000016    | 000 | (RW,I,GBL,ABS,OVR) |                     |                       |                    |     |
|                  | 001 | (RW,I,LCL,REL,CON) |                     |                       |                    |     |
| \$STATE 000152   | 002 | (RW,D,LCL,REL,CON) |                     |                       |                    |     |
| \$KTAB 000012    | 003 | (RW,D,LCL,REL,CON) |                     |                       |                    |     |
| \$KSTR 000023    | 004 | (RW,D,LCL,REL,CON) |                     |                       |                    |     |
| Errors detected: | 0   |                    |                     |                       |                    |     |

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10768 Words ( 43 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:29.48  
SY: TMPSTA.V2,[132,134]TMPSTA/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPSTA

|                  |                       |                   |                   |                      |
|------------------|-----------------------|-------------------|-------------------|----------------------|
| CS.SVC= ***** GX | SETOWN 000672R        | \$SFLG 000010     | \$SVRCL= ***** GX | \$GTR50= ***** GX    |
| DTE 000174R      | 002 SETPSZ 000514R    | \$SLEN 000032     | \$SVTIM= ***** GX | \$LAMDA= 000000      |
| DTE1 000176R     | 002 SETRCL 000610R    | \$SLENX= ***** GX | \$SVWSZ= ***** GX | \$NUMBR= 000002      |
| ERRDTE 000252R   | SETTIM 000630R        | \$SLFT 000012     | \$SWND 000006     | \$RAD50= 000016      |
| ERRFLG 000124R   | SETWSZ 000542R        | \$SLNK 000000     | \$ALPHA= 000022   | \$STRNG= 000004      |
| ERRNAM 000000R   | SF\$BLK= 000001       | \$SOWNR 000030    | \$ANY = 000020    | \$SUBXP= 000010      |
| ERROWN 000307R   | SF\$INC= 000002       | \$SPKSZ 000004    | \$BLANK= 000006   | \$SVUPD= ***** GX    |
| ERRPSZ 000032R   | SVCCTB 000000RG       | \$SRCL 000013     | \$BYTMN= ***** GX | \$TALOC= ***** GX    |
| ERRRCL 000153R   | SVCSTB 000000RG       | \$SSTA 000003     | \$CFERR= ***** GX | \$TMSVC 000336RG     |
| ERRTIM 000214R   | SVC1 000046R          | \$STIM 000014     | \$CLERR= ***** GX | \$\$\$\$FLG= 177777  |
| ERRWSZ 000067R   | SVC2 000066R          | \$STMR 000016     | \$DIGIT= 000024   | \$\$\$\$KEY= 000000  |
| EX\$SEV= 000004  | SV\$DTE= ***** GX     | \$VDTE= ***** GX  | \$DNUMB= 000014   | \$\$\$\$STA= 000000  |
| OPT1 000154R     | 002 SV\$PSZ= ***** GX | \$VFLG= ***** GX  | \$EOS = 000012    | \$\$\$\$TMP= 000000R |
| SETDTE 000650R   | SV\$WSZ= ***** GX     | \$VLEN= ***** GX  | \$ERROR= ***** GX | .ERROR= ***** GX     |
| SETFLG 000570R   | \$ADSZ 000007         | \$VNAM= ***** GX  | \$EXIT = 000000   | .FLAGS= ***** GX     |
| SETNM1 000454R   | \$SCTL 000002         | \$SVQWN= ***** GX | \$FAIL = 177777   | .TEMP = ***** GX     |
| SETNM2 000474R   | \$SDTE 000020         | \$VPSZ= ***** GX  | \$GTNUM= ***** GX | .TPARS= ***** GX     |

. ABS. 000032 000 (RW,I,GBL,ABS,OVR)  
000706 001 (RW,I,LCL,REL,CON)  
\$STATE 000206 002 (RW,D,LCL,REL,CON)  
\$KTAB 000002 003 (RW,D,LCL,REL,CON)  
\$KSTR 000004 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 11032 Words ( 44 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:37.42  
SY: TMPSVC.V2,[132,134]TMPSVC/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPSVC

```

253 .SBTTL SETCSR - SET THE SECONDARY CSR ADDRESS
254
255 +
256 *** - SETCSR - SET THE SECONDARY CSR ADDRESS
257
258 INPUT:
259 .PNUMB = SECONDARY CSR ADDRESS
260 .PNUMH = SECONDARY CSR ADDRESS (HIGH ORDER)
261 .TEMP = TEMPLATE ADDRESS
262
263 OUTPUT:
264 THE SECONDARY CSR ADDRESS IS STORED IN C.CSR OF THE TEMPLATE.
265
266 -
267
268 SETCSR:
269 000172 001700 000000G MOV .TEMP,RO ; GET START OF TEMPLATE
270 000176 016760 000000G 000012 MOV .PNUMB,C.CSR(RO) ; STORE THE SECONDARY CSR ADDRESS IN THE TEMPLATE
271 000204 052760 000000G 000000G BIS #UN$SCS,C.STS(RO) ; REMEMBER OPTION SET
272 000212 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
273 000216 001004 BNE 10$; IF NE, NO - REJECT
274 000220 022767 160000 000000G CMP #160000,.PNUMB ; IS THE SECONDARY CSR ADDRESS IN RANGE ?
275 000226 101402 BLOS 20$; IF LOS, YES
276 000230 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
277 000234 000207 10$: RETURN
278 20$:

```

```

159 .SBTTL TPARS STATE TABLES
160 :
161 : INITIALIZE TABLES
162 :
163 :
164 ISTAT$ X2PSTB, X2PKTB
165 :
166 : PROCESS 'X2P$DF'
167 :
168 STATES$
169 TRANS$ $LAMDA,, $CLERR,ERRLIN,.ERROR
170 STATES$; LINE ID
171 TRANS$!LINID
172 STATES$
173 TRANS$ '<','>'
174 :
175 STATES$
176 TRANS$ $LAMDA,, $CLERR,ERRWSZ,.ERROR
177 STATES$; MAXIMUM WINDOW BLOCK SIZE
178 TRANS$ $NUMBR,,SETWSZ
179 STATES$
180 TRANS$ '<','>'
181 :
182 STATES$
183 TRANS$ $LAMDA,, $CLERR,ERRBSZ,.ERROR
184 STATES$; MAXIMUM BLOCK SIZE
185 TRANS$ $NUMBR,,SETBSZ
186 STATES$
187 TRANS$ '<','>'
188 :
189 STATES$
190 TRANS$ $LAMDA,, $CLERR,ERRRTC,.ERROR
191 STATES$; MAXIMUM RETRANSMIT COUNT
192 TRANS$ $NUMBR,,SETRTC
193 STATES$
194 TRANS$ '<','>'
195 :
196 STATES$
197 TRANS$ $LAMDA,, $CLERR,ERRRTT,.ERROR
198 STATES$; RETRANSMIT TIMER
199 TRANS$ $NUMBR,,SETRTT
200 STATES$
201 TRANS$ '<','>'
202 :
203 STATES$
204 TRANS$ $LAMDA,, $CLERR,ERRHBT,.ERROR
205 STATES$; HOLDBACK TIMER
206 TRANS$ $NUMBR,,SETHBT
207 STATES$
208 TRANS$ $EOS,$EXIT
209 :
210 :
211 : SUBEXPRESSION TO PARSE LINE ID
212 :
213 STATES$ LINID ;DEVICE NAME
214 TRANS$ $STRNG,,SETDEV
215 STATES$

```

LOCAL DATA

```

126 .SBTTL LOCAL DATA
127
128 ; SYNTAX ERROR DETAIL STRINGS
129 ;
130 ;
131 .NLIST BEX
132 .EX$SEV = 4
133 000000 004 103 106 ERRMXC: .ASCIZ <EX$SEV>/CFE -- Error in maximum circuits/
134 000042 004 103 106 ERRCT: .ASCIZ <EX$SEV>/CFE -- Error in counter timer/
135 .EVEN
136 .LIST BEX

```

```

167 .SBTTL TPARS STATE TABLES
168 ;
169 ; INITIALIZE TABLES
170 ;
171 ;
172 000610 ISTAT$ X3PSTB, X3PKTB
173 ;
174 ; PROCESS 'X3P$DF'
175 ;
176 ;
177 000610 STATES$
178 000610 TRANS$ $LAMDA,, $CLERR,ERRDFP,,ERROR
179 000610 STATES$; DEFAULT BLOCK SIZE
180 000610 TRANS$ $NUMBR,,SETDFP
181 000610 STATES$
182 000610 TRANS$ <','>
183 ;
184 000610 STATES$
185 000610 TRANS$ $LAMDA,, $CLERR,ERRMXP,,ERROR
186 000610 STATES$; MAXIMUM BLOCK SIZE
187 000610 TRANS$ $NUMBR,,SETMXP
188 000610 STATES$
189 000610 TRANS$ <','>
190 ;
191 000610 STATES$
192 000610 TRANS$ $LAMDA,, $CLERR,ERRDW,,ERROR
193 000610 STATES$; DEFAULT WINDOW SIZE
194 000610 TRANS$ $NUMBR,,SETDW
195 000610 STATES$
196 000610 TRANS$ <','>
197 ;
198 000610 STATES$
199 000610 TRANS$ $LAMDA,, $CLERR,ERRMW,,ERROR
200 000610 STATES$; MAXIMUM WINDOW SIZE
201 000610 TRANS$ $NUMBR,,SETMW
202 000610 STATES$
203 000610 TRANS$ <','>
204 ;
205 000610 STATES$
206 000610 TRANS$ $LAMDA,, $CLERR,ERRCT,,ERROR
207 000610 STATES$; CALL TIMER VALUE
208 000610 TRANS$ $NUMBR,,SETCT
209 000610 STATES$
210 000610 TRANS$ <','>
211 ;
212 000610 STATES$
213 000610 TRANS$ $LAMDA,, $CLERR,ERRKT,,ERROR
214 000610 STATES$; CLEAR TIMER VALUE
215 000610 TRANS$ $NUMBR,,SETKT
216 000610 STATES$
217 000610 TRANS$ <','>
218 ;
219 000610 STATES$
220 000610 TRANS$ $LAMDA,, $CLERR,ERRRT,,ERROR
221 000610 STATES$; RESET TIMER VALUE
222 000610 TRANS$ $NUMBR,,SETRT
223 000610 STATES$

```



```
170 ; FINAL STATE
171 ;
172 000002 STATES
```

UPDCHN - UPDATE OUTGOING CHANNE MACRO V05.03b Saturday 29-Jun-85 05:47 <sup>J 14</sup> Page 6-1  
Symbol table

|                   |                   |                   |                   |                  |
|-------------------|-------------------|-------------------|-------------------|------------------|
| CHNDF = 000000R   | C\$HLO = ***** GX | \$OBUF = ***** GX | \$UPCHN 000010RG  | .COMMA= ***** GX |
| CHNLN = 000010    | F.NRBD= ***** GX  | \$OFDB = ***** GX | \$WDATA= ***** GX | .FLAGS= ***** GX |
| C\$HFI = ***** GX | \$BFI = ***** GX  | \$PTNUM= ***** GX | .BUFF = ***** GX  | .TEMP = ***** GX |
| C\$HHI = ***** GX | \$MVASC= ***** GX | \$PTOPT= ***** GX |                   |                  |

. ABS. 000000 000 (RW,I,GBL,ABS,OVR)  
000150 001 (RW,I,LCL,REL,CON)

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 8783 Words ( 35 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:05.06

SY:UPDCHN.V2,[132,134]UPDCHN/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDCHN

265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275  
276  
277  
278  
279  
280  
281  
282  
283  
284  
285  
286  
287  
288

000244  
000244 016702 177530  
000250 016201 000014  
000254 001406  
000256 016700 000000G  
000262 004767 000000G  
000266 010067 000000G  
000272 000207  
  
000001

```
.SBTTL SETURM - SET UNIBUS RUN MASK
*** - SETURM - SET UNIBUS RUN MASK
INPUTS:
 TEMP = TEMPLATE ADDRESS
 $OBUF = POINTER INTO OUTPUT BUFFER
OUTPUTS:
 THE UNIBUS RUN MASK IS STORED IN THE OUTPUT BUFFER IF NON-ZERO
-
SETURM:
 MOV TEMP,R2 ; GET THE TEMPLATE ADDRESS
 MOV C.URM(R2),R1 ; GET THE UNIBUS RUN MASK
 BEQ 10$, ; IF EQ, NONE
 MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
 CALL $CNV0 ; CONVERT NUMBER TO ASCII OCTAL
 MOV R0,$OBUF ; UPDATE POINTER TO OUTPUT BUFFER
10$: RETURN
.END
```

UPDCUG      CREATED BY    MACRO    ON 29-JUN-85 AT 05:48

PAGE 2      I 16

MACRO CROSS REFERENCE

CREF    04.00

MACRO NAME      REFERENCES

|           |        |       |       |
|-----------|--------|-------|-------|
| EPRINT    | #4-79  |       |       |
| ERROR\$   | #4-68  |       |       |
| PRINT     | #4-63  |       |       |
| \$PTNMB   | #4-112 | 6-178 |       |
| \$PTNUM   | #4-111 | 6-176 |       |
| \$PTRSO   | #4-116 | 6-175 |       |
| \$\$\$PTN | #4-91  | 4-111 | 4-112 |

IMPRDT CREATED BY MACRO ON 29-JUN-85 AT 05:40

PAGE 1

J 1

SYMBOL CROSS REFERENCE

CREF 04.00

| SYMBOL    | VALUE      | REFERENCES         |
|-----------|------------|--------------------|
| CS.RDT    | = ***** GX | 7-196              |
| ERRADR    | 000031 R   | #5-134             |
| ERRNAM    | 000000 R   | #5-133             |
| EX\$SEV   | = 000004   | #5-132 5-133 5-134 |
| RDTKTB    | 000000 RG  | #6-143 7-200       |
| RDTSTB    | 000000 RG  | #6-143 7-201       |
| RSDADR    | = ***** GX | 8-225              |
| RSDLEN    | = ***** GX | 7-191              |
| RSDNAM    | = ***** GX | 8-223              |
| R\$LENX   | = ***** GX | 7-198              |
| SETADR    | 000214 R   | #8-225             |
| SETNAM    | 000200 R   | #8-223             |
| \$ALPHA   | = 000022   | #6-143             |
| \$ANY     | = 000020   | #6-143             |
| \$BLANK   | = 000006   | #6-143             |
| \$BYTMN   | = ***** GX | *7-198             |
| \$CFERR   | = ***** GX | 7-206              |
| \$DIGIT   | = 000024   | #6-143             |
| \$DNUMB   | = 000014   | #6-143             |
| \$EOS     | = 000012   | #6-143             |
| \$ERROR   | = ***** GX | *7-207             |
| \$EXIT    | = 000000   | #6-143             |
| \$FAIL    | = 177777   | #6-143             |
| \$GPRM    | = *****    | 6-143              |
| \$GTR50   | = ***** GX | 8-223 8-225        |
| \$LAMDA   | = 000000   | #6-143             |
| \$NUMBR   | = 000002   | #6-143             |
| \$RAD50   | = 000016   | #6-143             |
| \$RONLY   | = *****    | 6-143 6-143        |
| \$STRNG   | = 000004   | #6-143             |
| \$SUBXP   | = 000010   | #6-143             |
| \$TALOC   | = ***** GX | 7-192              |
| \$TMRDT   | 000066 RG  | #7-189             |
| \$\$\$FLG | = 177777   | #6-143             |
| \$\$\$KEY | = 177777   | #6-143             |
| .ERROR    | = ***** GX | *7-197 7-204       |
| .FLAGS    | = ***** GX | *7-195             |
| .TEMP     | = ***** GX | *7-193             |
| .TPARS    | = ***** GX | 7-202              |

TMPREM      CREATED BY    MACRO    ON 29-JUN-85 AT 05:40      PAGE 1      J 2  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE       | REFERENCES                                                                             |
|-----------|-------------|----------------------------------------------------------------------------------------|
| CS.REM    | = ***** GX  | 6-148      7-186                                                                       |
| C.ADD     | = 000010    | #4-84      *6-147      7-180      7-188      7-188      7-192      *9-249      *10-274 |
| C.LEN     | = 000012    | #4-85      6-142                                                                       |
| C.NAM     | = 000004    | #4-83      *8-215      *8-220                                                          |
| C.STS     | = ***** GX  | *6-148      7-186                                                                      |
| FINREM    | = 000114 R  | 6-156      #7-177                                                                      |
| LA        | = 000074    | #4-73                                                                                  |
| RA        | = 000076    | #4-74                                                                                  |
| REMKTB    | = 000000 RG | #5-94      6-150                                                                       |
| REMSTB    | = 000000 RG | #5-94      6-151                                                                       |
| R.LENX    | = ***** GX  | 6-158                                                                                  |
| SETADD    | = 000324 R  | #10-268                                                                                |
| SETARE    | = 000262 R  | #9-239                                                                                 |
| SETNAM    | = 000202 R  | #8-210                                                                                 |
| TEMP      | = 000000 R  | #4-75      *6-144      6-155      8-214      8-219      9-245      10-273              |
| \$ALPHA   | = 000022    | #5-94                                                                                  |
| \$ANY     | = 000020    | #5-94                                                                                  |
| \$BLANK   | = 000006    | #5-94                                                                                  |
| \$BYTMN   | = ***** GX  | *6-158                                                                                 |
| \$CAT5    | = ***** GX  | 8-213      8-218                                                                       |
| \$DICIT   | = 000024    | #5-94                                                                                  |
| \$DNUMB   | = 000014    | #5-94                                                                                  |
| \$EOS     | = 000012    | #5-94                                                                                  |
| \$ERROR   | = ***** GX  | *6-154                                                                                 |
| \$EXIT    | = 000000    | #5-94                                                                                  |
| \$FAIL    | = 177777    | #5-94                                                                                  |
| \$GPRM    | = *****     | 5-94                                                                                   |
| \$LAMDA   | = 000000    | #5-94                                                                                  |
| \$LSNAD   | = ***** GX  | 7-180      *7-192                                                                      |
| \$LSTMP   | = ***** GX  | 7-178      *7-191                                                                      |
| \$NTADD   | = ***** GX  | 6-145                                                                                  |
| \$NUMBER  | = 000002    | #5-94                                                                                  |
| \$QINSN   | = ***** GX  | 6-157                                                                                  |
| \$RAD50   | = 000016    | #5-94                                                                                  |
| \$RONLY   | = *****     | 5-94      5-94                                                                         |
| \$STRNG   | = 000004    | #5-94                                                                                  |
| \$SUBXP   | = 000010    | #5-94                                                                                  |
| \$TMLST   | = ***** GX  | 7-182                                                                                  |
| \$TMREM   | = 000002 RG | #6-140                                                                                 |
| \$TIALC   | = ***** GX  | 6-143                                                                                  |
| \$\$\$FLG | = 177777    | #5-94                                                                                  |
| \$\$\$KEY | = 177777    | #5-94                                                                                  |
| .PNUMB    | = ***** GX  | 9-240      10-269                                                                      |
| .PSTCN    | = ***** GX  | 8-221                                                                                  |
| .PSTPT    | = ***** GX  | 8-211                                                                                  |
| .TPARS    | = ***** GX  | 6-152                                                                                  |

IMPROU      CREATED BY MACRO ON 29-JUN-85 AT 05:41      PAGE 2      J 3  
SYMBOL CROSS REFERENCE      CREF    04.00

SYMBOL    VALUE      REFERENCES

.TEMP    =   \*\*\*\*\*   GX   \*6-225  
.TPARS   =   \*\*\*\*\*   GX    6-233

```

352 .SBTTL SERDV - STORE SERVICE DEVICE
353 .SBTTL DADDR - STORE DUMP ADDRESS
354 .SBTTL DCT - STORE DUMP COUNT
355
356
357 ** SERDV - STORE SERVICE DEVICE
358
359 INPUT: TPARS CONVENTIONS
360
361 OUTPUT: THE SERVICE DEVICE IS STORE IN RADRO
362 AT S$DEV IN THE CURRENT TEMPLATE.
363
364 SERDV: MOV .PSTPT,RO ; GET STRING ADDRESS
365 CALL $CAT5 ; CONVERT TO RAD50
366 MOV .TEMP,RO
367 MOV R1,S$DEV(RO) ; STORE DEVICE NAME
368 RETURN
369
370
371 ** DADDR - STORE DUMP ADDRESS
372
373 INPUT: TPARS CONVENTIONS
374
375 OUTPUT: THE 32 BIT DUMP ADDRESS IS STORED
376 AT S$DPA IN THE CURRENT TEMPLATE.
377
378 DADDR: MOV .TEMP,RO
379 MOV .PNUMB,S$DPA(RO) ; STORE DUMP ADDRESS
380 MOV .PNUMH,S$DPA+2(RO) ; ...
381 RETURN
382
383
384 ** DCT - STORE DUMP COUNT
385
386 INPUT: TPARS CONVENTIONS
387
388 OUTPUT: THE DUMP COUNT IS STORED AT S$DPC
389 IN THE CURRENT TEMPLATE.
390
391 DCT: MOV .TEMP,RO
392 MOV .PNUMB,S$DPC(RO) ; STORE DUMP COUNT
393 RETURN
394

```



```

193
194 000376 STATES$
195 000376 TRANS$ $NUMBR,HTM,SETCTT
196 000376 TRANS$ $LAMDA
197
198 000376 STATES$ HTM ; OPTIONAL HELLO TIMER
199 000376 TRANS$ $EOS,EXIT
200 000376 TRANS$ <'>,$CLERR,ERRHTM,.ERROR
201
202 000376 STATES$
203 000376 TRANS$ $NUMBR,NBRA,SETHTM
204 000376 TRANS$ <'>
205
206 000376 STATES$ NBRA ; OPTIONAL MAX ROUTERS
207 000376 TRANS$ $EOS,EXIT
208 000376 TRANS$ <'>
209 000376 STATES$
210 000376 TRANS$ $NUMBR,RPRI,SETBRA
211 000376 TRANS$ <'>
212
213 000376 STATES$ RPRI ; OPTIONAL ROUTING PRIORITY
214 000376 TRANS$ $EOS,EXIT
215 000376 TRANS$ <'>
216 000376 STATES$
217 000376 TRANS$ $NUMBR,EXIT,SETRPR
218
219
220 000376 STATES$ LFS
221 000376 TRANS$ 'LF'
222
223 000376 STATES$
224 000376 TRANS$ ',$EXIT
225
226 000376 STATES$ SKIP
227 000376 TRANS$ '! ,NEXT
228 000376 TRANS$ <'>,$EXIT
229
230 000376 STATES$ NEXT
231 000376 TRANS$!LFS
232
233 000376 STATES$ BITS
234 000376 TRANS$ 'TIM',SKIP,,LF.TIM,FLAG
235 000376 TRANS$ 'MFL',SKIP,,LF.MFL,FLAG
236 000376 TRANS$ 'MDC',SKIP,,LF.MDC,FLAG
237 000376 TRANS$ 'MTP',SKIP,,LF.MTP,FLAG
238 000376 TRANS$ 'ENA',SKIP,,LF.ENA,FLAG
239 000376 TRANS$ 'X2P',SKIP,,LF.X2P,FLAG
240 000376 TRANS$ 'BRO',SKIP,,LF.BRO,FLAG
241 000376 TRANS$ 'SER',SKIP,,LF.SER,FLAG
242
243 ; EXIT STATE
244 ;
245 000376 STATES$ EXIT
246 000376 TRANS$ $LAMDA,$EXIT,SETFLG
247
248 000376 STATES$

```

\*\*FILE\*\*ID\*\*TMPSTA

J 6

```
TTTTTTTTTT MM MM PPPPPPP SSSSSSSS TTTTTTTTTT AAAAAA
TTTTTTTTTT MM MM PPPPPPP SSSSSSSS TTTTTTTTTT AAAAAA
 TT MM MM PP PP SS TT AA AA
 TT MMMM MMMM PP PP SS TT AA AA
 TT MM MM PP PP SS TT AA AA
 TT MM MM PP PP SS TT AA AA
 TT MM MM PPPPPPP SSSSSS TT AA AA
 TT MM MM PPPPPPP SSSSSS TT AA AA
 TT MM MM PP SS TT AAAAAAAAAA
 TT MM MM PP SS TT AAAAAAAAAA
 TT MM MM PP SS TT AA AA
 TT MM MM PP SS TT AA AA
 TT MM MM PP SSSSSSSS TT AA AA
 TT MM MM PP SSSSSSSS TT AA AA
 TT MM MM PP SSSSSSSS TT AA AA
```

```
LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT
```

TMPSTA      CREATED BY    MACRO    ON 29-JUN-85 AT 05:43      PAGE 1      J 7  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE      | REFERENCES                                              |
|-----------|------------|---------------------------------------------------------|
| CS.STA    | = ***** GX | 6-203                                                   |
| C.ADD     | 000005     | #4-112 *7-240                                           |
| C.APR     | 000010     | #4-115 *6-205 *9-293                                    |
| C.CST     | 000007     | #4-114 *6-204 *8-266                                    |
| C.FLG     | 000006     | #4-113 *8-267 *8-268                                    |
| C.HTM     | 000012     | #4-117 *10-319                                          |
| C.LEN     | = 000016   | #4-119 6-200                                            |
| C.STA     | 000004     | #4-111 *7-238                                           |
| C.STS     | = ***** GX | *6-203 *10-318                                          |
| ERRSTA    | 000004 R   | #4-101 7-236                                            |
| EX\$EV    | = 000004   | #4-100 4-101                                            |
| FLAG      | 000002 R   | #4-97 *6-199 8-268                                      |
| MXCSTL    | = ***** GX | 8-271                                                   |
| SETADD    | 000126 R   | #7-232                                                  |
| SETAPR    | 000266 R   | #9-291                                                  |
| SETCST    | 000216 R   | #8-264                                                  |
| SETHTM    | 000322 R   | #10-316                                                 |
| STAKTB    | 000000 RG  | #5-126 6-207                                            |
| STASTB    | 000000 RG  | #5-126 6-208                                            |
| ST\$HTM   | = ***** GX | 10-318                                                  |
| TEMP      | 000000 R   | #4-96 *6-202 7-233 8-265 9-292 10-317                   |
| \$ALPHA   | = 000022   | #5-126                                                  |
| \$ANY     | = 000020   | #5-126                                                  |
| \$BLANK   | = 000006   | #5-126                                                  |
| \$CFERR   | = ***** GX | 7-236                                                   |
| \$DIGIT   | = 000024   | #5-126                                                  |
| \$DNUMB   | = 000014   | #5-126                                                  |
| \$EOS     | = 000012   | #5-126                                                  |
| \$ERROR   | = ***** GX | *6-211                                                  |
| \$EXIT    | = 000000   | #5-126                                                  |
| \$FAIL    | = 177777   | #5-126                                                  |
| \$GPRM    | = *****    | 5-126                                                   |
| \$LAMDA   | = 000000   | #5-126                                                  |
| \$MXSTA   | = ***** GX | 7-234                                                   |
| \$NUMBER  | = 000002   | #5-126                                                  |
| \$RAD50   | = 000016   | #5-126                                                  |
| \$RONLY   | = *****    | 5-126 5-126 5-126                                       |
| \$STAN    | = ***** GX | 7-234 7-238 *7-239                                      |
| \$STRNG   | = 000004   | #5-126                                                  |
| \$SUBXP   | = 000010   | #5-126                                                  |
| \$TALOC   | = ***** GX | 6-201                                                   |
| \$TMSTA   | 000036 RG  | #6-197                                                  |
| \$\$\$FLG | = 177777   | #5-126                                                  |
| \$\$\$KEY | = 177777   | #5-126                                                  |
| .PNUMB    | = ***** GX | 7-240 7-241 8-266 8-269 9-294 10-320 9-293 9-296 10-319 |
| .PNUMH    | = ***** GX | 7-241                                                   |
| .TPARS    | = ***** GX | 6-209                                                   |

| SYMBOL   | VALUE      | REFERENCES |        |       |       |       |       |       |             |
|----------|------------|------------|--------|-------|-------|-------|-------|-------|-------------|
| CS.SVC   | = ***** GX | 7-254      |        |       |       |       |       |       |             |
| ERRDTE   | 000252 R   | #5-141     |        |       |       |       |       |       |             |
| ERRFLG   | 000124 R   | #5-138     |        |       |       |       |       |       |             |
| ERRNAM   | 000000 R   | #5-135     |        |       |       |       |       |       |             |
| ERROWN   | 000307 R   | #5-142     |        |       |       |       |       |       |             |
| ERRPSZ   | 000032 R   | #5-136     |        |       |       |       |       |       |             |
| ERRRCL   | 000153 R   | #5-139     |        |       |       |       |       |       |             |
| ERRTIM   | 000214 R   | #5-140     |        |       |       |       |       |       |             |
| ERRWSZ   | 000067 R   | #5-137     |        |       |       |       |       |       |             |
| EX\$SEV  | = 000004   | #5-134     | 5-135  | 5-136 | 5-137 | 5-138 | 5-139 | 5-140 | 5-141 5-142 |
| SETDTE   | 000650 R   | #8-309     |        |       |       |       |       |       |             |
| SETFLG   | 000570 R   | #8-303     |        |       |       |       |       |       |             |
| SETNM1   | 000454 R   | #8-295     |        |       |       |       |       |       |             |
| SETNM2   | 000474 R   | #8-297     |        |       |       |       |       |       |             |
| SETOWN   | 000672 R   | #8-311     |        |       |       |       |       |       |             |
| SETPSZ   | 000514 R   | #8-299     |        |       |       |       |       |       |             |
| SETRCL   | 000610 R   | #8-305     |        |       |       |       |       |       |             |
| SETTIM   | 000630 R   | #8-307     |        |       |       |       |       |       |             |
| SETWSZ   | 000542 R   | #8-301     |        |       |       |       |       |       |             |
| SVCKTB   | 000000 RG  | #6-152     | 7-258  |       |       |       |       |       |             |
| SVCSTB   | 000000 RG  | #6-152     | 7-259  |       |       |       |       |       |             |
| SV\$DTE  | = ***** GX | 8-309      |        |       |       |       |       |       |             |
| SV\$PSZ  | = ***** GX | 8-299      |        |       |       |       |       |       |             |
| SV\$WSZ  | = ***** GX | 8-301      |        |       |       |       |       |       |             |
| \$LENX   | = ***** GX | 7-255      |        |       |       |       |       |       |             |
| \$VDTE   | = ***** GX | 8-309      |        |       |       |       |       |       |             |
| \$VFLG   | = ***** GX | 8-303      |        |       |       |       |       |       |             |
| \$VLEN   | = ***** GX | 7-249      |        |       |       |       |       |       |             |
| \$VNAM   | = ***** GX | 8-295      | 8-297  |       |       |       |       |       |             |
| \$VOWN   | = ***** GX | 8-311      |        |       |       |       |       |       |             |
| \$VPSZ   | = ***** GX | 8-299      |        |       |       |       |       |       |             |
| \$VRCL   | = ***** GX | 8-305      |        |       |       |       |       |       |             |
| \$VTIM   | = ***** GX | 8-307      |        |       |       |       |       |       |             |
| \$VWSZ   | = ***** GX | 8-301      |        |       |       |       |       |       |             |
| \$ALPHA  | = 000022   | #6-152     |        |       |       |       |       |       |             |
| \$ANY    | = 000020   | #6-152     |        |       |       |       |       |       |             |
| \$BLANK  | = 000006   | #6-152     |        |       |       |       |       |       |             |
| \$BYTMN  | = ***** GX | *7-255     |        |       |       |       |       |       |             |
| \$CFERR  | = ***** GX | 7-264      |        |       |       |       |       |       |             |
| \$DIGIT  | = 000024   | #6-152     |        |       |       |       |       |       |             |
| \$DNUMB  | = 000014   | #6-152     |        |       |       |       |       |       |             |
| \$EOS    | = 000012   | #6-152     |        |       |       |       |       |       |             |
| \$ERROR  | = ***** GX | *7-265     | *7-266 |       |       |       |       |       |             |
| \$EXIT   | = 000000   | #6-152     |        |       |       |       |       |       |             |
| \$FAIL   | = 177777   | #6-152     |        |       |       |       |       |       |             |
| \$GPRM   | = *****    | 6-152      |        |       |       |       |       |       |             |
| \$GTR50  | = ***** GX | 8-295      | 8-297  | 8-299 | 8-301 | 8-303 | 8-305 | 8-307 |             |
| \$LAMBDA | = 000000   | 8-309      | 8-311  |       |       |       |       |       |             |
| \$NUMBR  | = 000002   | #6-152     |        |       |       |       |       |       |             |
| \$RAD50  | = 000016   | #6-152     |        |       |       |       |       |       |             |
| \$RONLY  | = *****    | 6-152      | 6-152  | 6-152 |       |       |       |       |             |

```

278 .SBTTL SETCST - SET THE LINE COST
279 +
280 *** - SETCST - SET THE LINE COST
281 :
282 : INPUT:
283 : .PNUMB = LINE COST
284 : .PNUMH = LINE COST (HIGH ORDER)
285 : .TEMP = TEMPLATE ADDRESS
286 :
287 : OUTPUT:
288 : THE LINE COST IS STORED IN C.CST OF THE TEMPLATE.
289 :
290 : -
291
292 SETCST:
293 000236 016700 000000G MOV .TEMP,R0 ; GET START OF TEMPLATE
294 000242 016760 000000G 000014 MOV .PNUMB,C.CST(R0) ; STORE THE LINE COST IN THE TEMPLATE
295 000250 052760 000000G 000000G BIS #UN$CST,C.STS(R0) ; REMEMBER OPTION
296 000256 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
297 000262 001004 BNE 10$; IF NE, NO - REJECT
298 000264 026727 000000G 000000G CMP .PNUMB,#MXCSTL ; IS THE LINE COST IN RANGE ?
299 000272 101402 BLOS 20$; IF LOS, YES
300 000274 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
301 000300 000207 20$: RETURN

```

```

216 000276 TRANS <'->
217 000276 STATES
218 000276 TRANS $DNUMB,,SETCON ;CONTROLLER NUMBER
219 000276 STATES
220 000276 TRANS <'->,LINO
221 000276 TRANS $LAMDA,$EXIT
222 000276 STATES LINO
223 000276 TRANS $DNUMB,$EXIT,SETUNI ;OPTIONAL UNIT NUMBER
224
225 ; FINAL STATE
226 ;
227 000276 STATES
228

```

```

138 .SBTTL TPARS STATE TABLES
139 ;
140 ; INITIALIZE TABLES
141 ;
142 ;
143 ISTAT$ X29STB, X29KTb
144 ;
145 ; PROCESS 'X29$DF'
146 ;
147 ;
148 000102 STATES$
149 000102 TRANS$ $LAMDA,, $CLERR,ERRMXC,.ERROR
150 000102 STATES$; MAXIMUM CIRCUITS
151 000102 TRANS$ $NUMBR,,SETMXC
152 000102 STATES$
153 000102 TRANS$ <','>,$CLERR,ERRCT,.ERROR
154 ;
155 000102 STATES$
156 000102 TRANS$ $NUMBR,,SETCT ; COUNTER TIMER
157 ;
158 000102 STATES$
159 000102 TRANS$ $EOS,$EXIT
160 ;
161 ;
162 ; FINAL STATE
163 ;
164 000102 STATES$
165 ;

```

```

224 000610 TRANS <','>
225
226 000610 STATES$
227 000610 TRANS $LAMDA,, $CLERR,ERRST,.ERROR
228 000610 STATES$; RESTART TIMER VALUE
229 000610 TRANS $NUMBR,,SETST
230 000610 STATES$
231 000610 TRANS <','>
232
233 000610 STATES$
234 000610 TRANS $LAMDA,, $CLERR,ERRKM,.ERROR
235 000610 STATES$; MAXIMUM CLEAR VALUE
236 000610 TRANS $NUMBR,,SETKM
237 000610 STATES$
238 000610 TRANS <','>
239
240 000610 STATES$
241 000610 TRANS $LAMDA,, $CLERR,ERRRM,.ERROR
242 000610 STATES$; MAXIMUM RESETS VALUE
243 000610 TRANS $NUMBR,,SETRM
244 000610 STATES$
245 000610 TRANS <','>
246
247 000610 STATES$
248 000610 TRANS $LAMDA,, $CLERR,ERRSM,.ERROR
249 000610 STATES$; MAXIMUM RESTARTS VALUE
250 000610 TRANS $NUMBR,,SETSM
251
252 000610 STATES$
253 000610 TRANS $EOS,$EXIT
254
255 ;
256 ; FINAL STATE
257 ;
258 000610 STATES$
259

```



```

174 .SBTTL $UPBUF - UPDATE BUFFER TEMPLATE
175 ;
176 *** - $UPBUF - UPDATE BUFFER TEMPLATE
177 ;
178 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
179 ;
180 INPUT:
181 R3-R5 - TPARS REGISTERS
182 ;
183 OUTPUT:
184 C-BIT = SUCCESS/FAILURE
185 ;
186 ;
187 ;
188 $UPBUF::
189 000002 012700 000000G MOV #TMLST,R0 ; POINT AT TEMPLATE LIST
190 000006 010001 10$: MOV R0,R1 ; SAVE CURRENT TEMPLATE
191 000010 011000 MOV (R0),R0 ; GET NEXT TEMPLATE
192 000012 001441 BEQ 10$, ; IF EQ, END OF LIST
193 000014 122760 000000G 000000G CMPB #CS.BUF,C.STS(R0) ; IS THIS THE BUFFER TEMPLATE ?
194 000022 001371 BNE 10$, ; IF NE, NO .. KEEP LOOKING
195 ;
196 ; ASSUME THIS IS THE RIGHT TEMPLATE
197 ;
198 000024 011011 MOV (R0),(R1) ; REMOVE FROM LIST
199 000026 010067 177746 MOV R0,NEXT ; SAVE TEMPLATE ADDRESS
200 000032 032760 000000G 000000G BIT #CS.MOD,C.STS(R0) ; WAS THIS TEMPLATE MODIFIED ?
201 000040 001004 BNE 15$, ; IF NE, YES .. UPDATE THE RECORD
202 000042 012767 000000G 000000G MOV #SBUFF,$OFDB+F.NRBD+2 ; OUTPUT THE INPUT BUFFER
203 000050 000421 BR 30$, ; AND LEAVE
204 000052 062767 000004 177720 15$: ADD #C.CBNM,NEXT ; POINT AT FIRST LOCATION IN TEMPLATE
205 000060 004767 000000G CALL $MOVST ; MOVE FIRST FIELD INTO OUTPUT BUFFER
206 000064 010546 MOV R5,-(SP) ; SAVE R5
207 000066 005001 CLR R1 ; IGNORE BLANKS
208 000070 012702 000000' MOV #BUFKTB,R2 ; KEYWORD TABLE
209 000074 012705 000000' MOV #BUFSTB,R5 ; STATE TABLE
210 000100 004767 000000G CALL .TPARS ; PARSE THE REST OF THE LINE
211 000104 103002 BCC 20$, ; NORMAL RETURN IF NO ERROR
212 000106 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
213 000112 012605 20$: MOV (SP)+,R5 ; RESTORE R5
214 000114 000207 30$: RETURN
215 ;
216 ; ERROR CONDITIONS
217 ;
218 000116 000167 000000G 101$: JMP .CONER ; CONSISTENCY ERROR

```

UPDCHN      CREATED BY    MACRO    ON 29-JUN-85 AT 05:47      PAGE 1      J 14

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE      | REFERENCES                     |
|---------|------------|--------------------------------|
| CHNDF   | 000000 R   | #5-126      5-127      6-151   |
| CHNLN   | = 000010   | #5-127      6-152              |
| CH\$HI  | = ***** GX | 6-156                          |
| C\$HHI  | = ***** GX | 6-156                          |
| C\$HLO  | = ***** GX | 6-155                          |
| F.NRBD  | = ***** GX | *6-145      *6-159      *6-161 |
| \$BUFI  | = ***** GX | 6-161                          |
| \$MVASC | = ***** GX | 6-153                          |
| \$OBUF  | = ***** GX | *6-154      6-157              |
| \$OFDB  | = ***** GX | *6-145      *6-159      *6-161 |
| \$PTNUM | = ***** GX | 6-155      6-156               |
| \$PTOPT | = ***** GX | 6-156                          |
| \$UPCHN | 000010 RG  | #6-144                         |
| \$WDATA | = ***** GX | 6-160                          |
| .BUFF   | = ***** GX | 6-145      6-150      6-158    |
| .COMMA  | = ***** GX | *6-149                         |
| .FLAGS  | = ***** GX | *6-148                         |
| .TEMP   | = ***** GX | *6-146                         |

UPDCNT - CFE UPDATE CONTROLLER MACRO V05.03b Saturday 29-Jun-85 05:48 Page 11-1  
 Symbol table

|                  |     |                  |                   |                   |                   |
|------------------|-----|------------------|-------------------|-------------------|-------------------|
| CNTKTB 000000RG  | 003 | C.VEC 000006     | \$ANY = 000020    | \$FAIL = 177777   | \$SUBXP= 000010   |
| CNTSTB 000000RG  | 002 | F.NRBD= ***** GX | \$BLANK= 000006   | \$LAMDA= 000000   | \$TMLST= ***** GX |
| CS.CNT= ***** GX |     | RCDEV 000052R    | \$BUFI = ***** GX | \$MOVPT= ***** GX | \$UPCNT 000002RG  |
| CS.MOD= ***** GX |     | SETCNT 000114R   | \$CLNUP= ***** GX | \$MOVST= ***** GX | \$\$\$FLG= 177777 |
| C.CNT 000004     |     | SETCSR 000170R   | \$CNVO = ***** GX | \$NUMBR= 000002   | \$\$\$KEY= 177777 |
| C.CSR 000010     |     | SETPRI 000216R   | \$DIGIT= 000024   | \$OBUF = ***** GX | \$\$\$STA= 000000 |
| C.LEN = 000016   |     | SETURM 000244R   | \$DNUMB= 000014   | \$OFDB = ***** GX | .CONER= ***** GX  |
| C.PRI 000012     |     | SETVEC 000142R   | \$EOS = 000012    | \$RAD50= 000016   | .PNUMB= ***** GX  |
| C.STS = ***** GX |     | TEMP 000000R     | \$ERROR= ***** GX | \$STRNG= 000004   | .TPARS= ***** GX  |
| C.URM 000014     |     | \$ALPHA= 000022  | \$EXIT = 000000   |                   |                   |

. ABS. 000016 000 (RW,I,GBL,ABS,OVR)  
 000274 001 (RW,I,LCL,REL,CON)  
 \$STATE 000074 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000000 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000000 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

### \*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10406 Words ( 41 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:22.47  
 SY:UPDCNT.V2,[132,134]UPDCNT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDCNT

|            |    |         |          |          |          |      |      |      |
|------------|----|---------|----------|----------|----------|------|------|------|
| UU         | UU | PPPPPPP | DDDDDDDD | DDDDDDDD | DDDDDDDD | MM   | MM   |      |
| UU         | UU | PPPPPPF | DDDDDDDD | DDDDDDDD | DDDDDDDD | MM   | MM   |      |
| UU         | UU | PP      | DD       | DD       | DD       | MMMM | MMMM |      |
| UU         | UU | PP      | DD       | DD       | DD       | MMMM | MMMM |      |
| UU         | UU | PP      | DD       | DD       | DD       | MM   | MM   |      |
| UU         | UU | PP      | DD       | DD       | DD       | MM   | MM   |      |
| UU         | UU | PP      | DD       | DD       | DD       | MM   | MM   |      |
| UU         | UU | PPPPPPP | DD       | DD       | DD       | MM   | MM   |      |
| UU         | UU | PPPPPPP | DD       | DD       | DD       | MM   | MM   |      |
| UU         | UU | PP      | DD       | DD       | DD       | MM   | MM   |      |
| UU         | UU | PP      | DD       | DD       | DD       | MM   | MM   |      |
| UU         | UU | PP      | DD       | DD       | DD       | MM   | MM   |      |
| UU         | UU | PP      | DD       | DD       | DD       | MM   | MM   |      |
| UU         | UU | PP      | DD       | DD       | DD       | MM   | MM   |      |
| UUUUUUUUUU | PP |         | DDDDDDDD | DDDDDDDD | DDDDDDDD | MM   | MM   | .... |
| UUUUUUUUUU | PP |         | DDDDDDDD | DDDDDDDD | DDDDDDDD | MM   | MM   | .... |
|            |    |         |          |          |          |      |      | .... |
|            |    |         |          |          |          |      |      | .... |

|            |          |           |
|------------|----------|-----------|
| LL         | SSSSSSSS | TTTTTTTTT |
| LL         | SSSSSSSS | TTTTTTTTT |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SSSSSS   | TT        |
| LL         | SSSSSS   | TT        |
| LL         |          | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LLLLLLLLLL | SSSSSSSS | TT        |
| LLLLLLLLLL | SSSSSSSS | TT        |

TMPRDT      CREATED BY    MACRO    ON 29-JUN-85 AT 05:40      PAGE 2      K 1

MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| CALLR   | #4-124 |        |        |        |        |        |        |        |        |        |
| DBGTP\$ | #6-143 | #6-167 |        |        |        |        |        |        |        |        |
| EPRINT  | #4-78  |        |        |        |        |        |        |        |        |        |
| ERROR\$ | #4-67  | 7-206  |        |        |        |        |        |        |        |        |
| ISTAT\$ | #4-124 | 6-143  |        |        |        |        |        |        |        |        |
| MTRANS  | #6-143 |        |        |        |        |        |        |        |        |        |
| PRINT   | #4-62  |        |        |        |        |        |        |        |        |        |
| STAT\$  | #4-124 | 6-147  | #6-149 | #6-151 | #6-154 | #6-156 | #6-158 | #6-163 | #6-165 | #6-171 |
| TRANS   | #4-124 | #6-148 | #6-150 | #6-152 | #6-155 | #6-157 | #6-159 | #6-164 | #6-166 | #6-167 |
| \$GTNMB | #4-100 |        |        |        |        |        |        |        |        |        |
| \$GTNUM | #4-90  |        |        |        |        |        |        |        |        |        |
| \$GTR50 | #4-112 | 8-223  | 8-225  |        |        |        |        |        |        |        |

TMPPREM      CREATED BY MACRO ON 29-JUN-85 AT 05:40      PAGE 2      K 2  
MACRO CROSS REFERENCE      CREF 04.00

MACRO NAME      REFERENCES

|         |       |        |        |        |        |        |        |        |        |        |
|---------|-------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$ | #5-94 | #5-111 |        |        |        |        |        |        |        |        |
| ISIAT\$ | #4-66 | 5-94   |        |        |        |        |        |        |        |        |
| MTRAN\$ | #5-94 |        |        |        |        |        |        |        |        |        |
| PRINT   | #4-57 |        |        |        |        |        |        |        |        |        |
| STATE\$ | #4-66 | 5-97   | #5-100 | #5-103 | #5-106 | #5-109 | #5-114 | #5-117 | #5-122 | #5-125 |
| TRAN\$  | #4-66 | #5-98  | #5-101 | #5-104 | #5-107 | #5-110 | #5-111 | #5-115 | #5-118 | #5-123 |

IMPROU      CREATED BY MACRO ON 29-JUN-85 AT 05:41      PAGE 3      K 3  
 MACRO CROSS REFERENCE      CREF      04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| CALLR   | #4-112 |        |        |        |        |        |        |        |        |        |
| DBGTP\$ | #5-147 | #5-167 | #5-174 | #5-204 |        |        |        |        |        |        |
| ERROR\$ | #4-63  | 6-237  |        |        |        |        |        |        |        |        |
| ISTAT\$ | #4-112 | 5-147  |        |        |        |        |        |        |        |        |
| MTRAN\$ | #5-147 |        |        |        |        |        |        |        |        |        |
| PRINT   | #4-58  |        |        |        |        |        |        |        |        |        |
| STATE\$ | #4-112 | 5-150  | #5-153 | #5-156 | #5-159 | #5-162 | #5-165 | #5-169 | #5-172 | #5-175 |
|         | #5-178 | #5-180 | #5-183 | #5-185 | #5-188 | #5-190 | #5-193 | #5-195 | #5-202 | #5-206 |
| TRANS   | #4-112 | #5-151 | #5-154 | #5-157 | #5-160 | #5-163 | #5-166 | #5-167 | #5-170 | #5-173 |
|         | #5-174 | #5-176 | #5-179 | #5-181 | #5-184 | #5-186 | #5-189 | #5-191 | #5-194 | #5-196 |
|         | #5-203 | #5-204 |        |        |        |        |        |        |        |        |
| \$GTNMB | #4-87  |        |        |        |        |        |        |        |        |        |
| \$GTNUM | #4-77  | 7-247  | 7-250  | 7-253  | 7-256  | 7-259  | 7-262  | 7-265  | 7-268  | 7-271  |
| \$GTR50 | #4-99  |        |        |        |        |        |        |        |        |        |

```

396 .SBTTL PAS - SERVICE BLOCK PASSWORD
397 .SBTTL STHAD - STORE HARDWARE ADDRESS
398
399
400 +
401 *** PAS - SERVICE BLOCK PASSWORD
402 *** STHAD - STORE HARDWARE ADDRESS
403
404 INPUTS -
405 .PSTPT - ADDRESS OF PARSED STRING
406 .PSTCN - SIZE OF STRING
407 .TEMP - ADDRESS OF TEMPLATE
408
409 OUTPUTS -
410 $SPSS - CONTAINS SERVICE BLOCK PASSWORD IN RAD50
411 $SHAD - CONTAINS HARDWARE ADDRESS IN RAD50
412
413 -
414 .ENABL LSB
415
416 STHAD: MOV $SHAD,R0 ; POINT TO HARDWARE ADDRESS SAVE AREA
417 MOV #12,-(SP) ; PUSH MAXIMUM STRING LENGTH
418 BR 5$; JOIN COMMON CODE
419
420 PAS: MOV $SPSS,R0 ; POINT TO PASSWORD SAVE AREA
421 MOV #16,-(SP) ; PUSH MAXIMUM STRING LENGTH
422
423 5$: MOV .PSTCN,R2 ; GET STRING COUNT
424 CMP R2,(SP)+ ; STRING TOO LONG ?
425 BHI 40$; IF HI, YES - REJECT TRANSITION
426
427 ; CHECK FOR INVALID CHARACTERS
428 ;
429 ;
430 10$: MOV .PSTPT,R1 ; POINT TO STRING
431 CMPB (R1),#'A ; IS IT ALPHABETIC ?
432 BLO 15$; IF LO, NO - TRY NUMERIC
433 CMPB (R1),#'Z ; IS IT REALLY ALPHABETIC ?
434 BLO 20$; IF LO, YES - VALID CHAR
435 BR 40$; ELSE REJECT TRANSITION
436
437 15$: CMPB (R1),#'0 ; IS IT NUMERIC ?
438 BLO 40$; IF LO, REJECT
439 CMPB (R1),#'9 ; NUMERIC ?
440 BHI 40$; IF HI, REJECT
441
442 20$: INC R1 ; NEXT CHAR
443 SOB R2,10$; ...
444
445 ; CONVERT TO RAD50
446 ;
447 ;
448 SAVRG R3 ; SAVE REGISTER
449 MOV R0,R3 ; PICK UP OFFSET FOR TARGET
450 ADD .TEMP,R3 ; POINT INTO TEMPLATE
451 MOV .PSTPT,R0 ; POINT TO PARSED STRING
452 MOVB (R4),-(SP) ; SAVE NEXT CHARACTER
453 MOVB #<'?',(R4) ; INSERT STOP CHARACTER
454 30$: CALL .CAT5E ; CONVERT TO RAD50 (EXTENDED WITH '"-')
455 BCC 33$; IF CC, OK
456 MOV R1,R1 ; ELSE ANYTHING GOOD ? (PRESERVE CARRY)

```



```

250 .SBTTL $TMSLT - SETUP SLT TEMPLATE
251 ;+
252 *** - $TMSLT - SETUP SLT TEMPLATE
253 ;
254 INPUT:
255 R3-R5 - TPARS REGISTERS
256 ;
257 OUTPUT:
258 C-BIT = SUCCESS/FAILURE
259 IF SUCCESS, THE SLT TEMPLATE IS STORED IN THE END OF TASK BUFFER.
260 ;
261 ;
262 ;
263 $TMSLT::
264 000376 010546 MOV R5, -(SP) ; SAVE R5
265 000376 005067 000000G CLR $STAN ; RESET STARTING STATION NUMBER (NEW SLT)
266 000400 005067 177370 CLR FLAG ; AND LINE FLAGS
267 000410 012701 000000G MOV #$$LLEN, R1 ; GET LENGTH OF ALLOCATION
268 000414 004767 000000G CALL $TALOC ; TRY TO ALLOCATE CORE BLOCK
269 000420 010067 000000G MOV R0, TEMP ; STORE TEMPLATE ADDRESS
270 000424 005720 TST (R0)+ ; POINT TO C.STS
271 000426 010067 000000G MOV R0, .FLAGS ; SAVE FOR OPTION SETTING
272 000432 012710 000000G MOV #CS.SLT, (R0) ; INDICATE THAT THIS IS A SLT$DF TEMPLATE
273 000436 005001 CLR R1 ; IGNORE BLANKS
274 000440 012702 000000' MOV #SLTKTB, R2 ; KEYWORD TABLE
275 000444 012705 000000' MOV #SLTSTB, R5 ; STATE TABLE
276 000450 004767 000000G CALL .TPARS ; PARSE THE REST OF THE LINE
277 000454 103012 BCC 20$; NORMAL RETURN IF NO ERROR
278 000456 016700 000000G MOV .ERROR, R0 ; GET ERROR DETAIL
279 000462 001405 BEQ 15$; IF EQ, NONE
280 000464 ERROR$ R0 ; PRINT DETAIL
281 000470 012767 177776 000000G MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
282 000476 005267 000000G 15$: INC $ERROR ; INDICATE SYNTAX ERROR
283 000502 012605 MOV (SP)+, R5 ; RESTORE R5
284 000504 000207 RETURN

```

TMPSTA - CFE PARSE STATION DEFINI MACRO V05.03b Saturday 29-Jun-85 05:43 <sup>K 6</sup>  
Table of contents

|     |     |                                             |
|-----|-----|---------------------------------------------|
| 6-  | 184 | \$TMSTA - SETUP STATION TEMPLATE            |
| 7-  | 215 | SETADD - SETUP STATION ADDRESS              |
| 8-  | 248 | SETCST - SETUP STATION COST AND FLAGS       |
| 9-  | 276 | SETAPR - SETUP STATION ACTIVE POLLING RATIO |
| 10- | 301 | SETHTM - SETUP STATION HELLO TIMER          |

TMPSTA      CREATED BY    MACRO    ON 29-JUN-85 AT 05:43      PAGE 2    K 7  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-126     | #5-139 | #5-143 | #5-144 | #5-151 | #5-158 | #5-174 | #5-178 | #5-179 | #5-180 |
| EPRINT     | #4-76      |        |        |        |        |        |        |        |        |        |
| ERROR\$    | #4-65      | 7-236  |        |        |        |        |        |        |        |        |
| ISTAT\$    | #4-90      | 5-126  |        |        |        |        |        |        |        |        |
| MTRANS     | #5-126     |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-60      |        |        |        |        |        |        |        |        |        |
| STAT\$     | #4-90      | 5-128  | #5-131 | #5-134 | #5-137 | #5-141 | #5-146 | #5-149 | #5-153 | #5-156 |
|            | #5-160     | #5-166 | #5-169 | #5-172 | #5-176 | #5-182 |        |        |        |        |
| TRANS      | #4-90      | #5-129 | #5-132 | #5-135 | #5-138 | #5-139 | #5-142 | #5-143 | #5-144 | #5-147 |
|            | #5-150     | #5-151 | #5-154 | #5-157 | #5-158 | #5-161 | #5-167 | #5-170 | #5-173 | #5-174 |
|            | #5-177     | #5-178 | #5-179 | #5-180 |        |        |        |        |        |        |

TMPSVC      CREATED BY MACRO ON 29-JUN-85 AT 05:44      PAGE 2      K 8

SYMBOL CROSS REFERENCE      CREF      04.00

| SYMBOL    | VALUE       | REFERENCES               |
|-----------|-------------|--------------------------|
| \$STRNG   | = 000004    | #6-152                   |
| \$SUBXP   | = 000010    | #6-152                   |
| \$SVUPD   | = ***** GX  | *7-247                   |
| \$TALOC   | = ***** GX  | 7-250                    |
| \$TMSVC   | = 000336 RG | #7-246                   |
| \$\$\$FLG | = 177777    | #6-152                   |
| \$\$\$KEY | = 177777    | #6-152                   |
| .ERROR    | = ***** GX  | *7-256 7-262             |
| .FLAGS    | = ***** GX  | *7-253 8-299 8-301 8-309 |
| .TEMP     | = ***** GX  | *7-251                   |
| .TPARS    | = ***** GX  | 7-260                    |

IMPUNT - CFE PARSE UNIT DEFINIT MACRO V05.03b Saturday 29-Jun-85 05:44 Page 12  
 SETDPR - SET THE LINE DEAD POLLING RATIO

```

303 .SBTTL SETDPR - SET THE LINE DEAD POLLING RATIO
304 ;+
305 *** - SETDPR - SET THE LINE DEAD POLLING RATIO
306 ;
307 INPUT:
308 .PNUMB = LINE DEAD POLLING RATIO
309 .PNUMH = LINE DEAD POLLING RATIO (HIGH ORDER)
310 .TEMP = TEMPLATE ADDRESS
311 ;
312 OUTPUT:
313 THE LINE DEAD POLLING RATIO IS STORED IN C.DPR OF THE TEMPLATE.
314 ;
315 ;
316 ;
317 SETDPR:
318 000302 016700 000000G MOV .TEMP,R0 ; GET START OF TEMPLATE
319 000306 016760 000000G 000016 MOV .PNUMB,C.DPR(R0) ; STORE THE LINE DEAD RATIO IN THE TEMPLATE
320 000314 052760 000000G 000000G BIS #UNSDPR,C.STS(R0) ; REMEMBER OPTION
321 000322 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
322 000326 001003 BNE 10$; IF NE, NO - REJECT
323 000330 105767 000001G TSTB .PNUMB+1 ; IS THE DEAD POLLING RATIO IN RANGE ?
324 000334 001402 BEQ 20$; IF EQ, YES
325 000336 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
326 000342 000207 20$: RETURN

```

IMPUNT - CFE PARSE UNIT DEFINIT MACRO V05.03b Saturday 29-Jun-85 05:44 Page 13  
 SETDPR - SET THE PROTOCOL EMULATOR CHARACTERISTICS

```

230 .SBTTL $TMX2P - BUILD X2P TEMPLATE
231 ;+
232 *** - $TMX2P - BUILD X2P TEMPLATE
233
234 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
235
236 INPUT:
237 R3-R5 - TPARS REGISTERS
238
239 OUTPUT:
240 C-BIT = SUCCESS/FAILURE
241 THE CHANNEL TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
242
243 ;+
244 ;+
245 $TMX2P::
246 MOV R5, -(SP) ; SAVE R5
247 MOV #C.LEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
248 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
249 MOV R0, .TEMP ; SAVE TEMPLATE ADDRESS
250 TST (R0)+ ; POINT TO C.STS
251 MOV R0, .FLAGS ; SAVE FOR OPTION SETTING
252 MOV #CS.X2P, (R0) ; INDICATE THAT THIS IS A X2P$DF TEMPLATE
253 CLR C.LIN+4(R0) ; CLEAR OPTIONAL CONTROLLER NUMBER
254 CLR .ERROR ; NO ERROR DETAIL YET
255 CLR R1 ; IGNORE BLANKS
256 MOV #X2PKTB, R2 ; KEYWORD TABLE
257 MOV #X2PSTB, R5 ; STATE TABLE
258 CALL .TPARS ; PARSE THE REST OF THE LINE
259 BCC 20$; NORMAL RETURN IF NO ERROR
260 MOV .ERROR, R0 ; CHECK FOR ERROR DETAIL TEXT
261 BEQ 15$; IF EQ, NO DETAIL TEXT
262 MOV R0, .ERROR$; PRINT ACTION ROUTINE ERROR
263 MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
264 INC $ERROR ; INDICATE SYNTAX ERROR
265 MOV (SP)+, R5 ; RESTORE R5
266 RETURN
267
246 000276 010546
247 000300 012701 000024
248 000304 004767 000000G
249 000310 010067 000000G
250 000314 005720
251 000316 010067 000000G
252 000322 012710 000000G
253 000326 005060 000010
254 000332 005067 000000G
255 000336 005001
256 000340 012702 000000'
257 000344 012705 000000'
258 000350 004767 000000G
259 000354 103012
260 000356 016700 000000G
261 000362 001405
262 000364
263 000370 012767 177776 000000G
264 000376 005267 15$:
265 000402 012605 20$:
266 000404 000207

```

```

167 .SBTTL $TMX29 - BUILD X29 TEMPLATE
168 +
169 *** - $TMX29 - BUILD X29 TEMPLATE
170 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
171 INPUT:
172 R3-R5 - TPARS REGISTERS
173 OUTPUT:
174 C-BIT = SUCCESS/FAILURE
175 THE X29 TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
176 -
177
178
179
180
181
182 000102 $TMX29::
183 000102 010546 MOV R5, -(SP) ; SAVE R5
184 000104 012701 000000G MOV #X$9LEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
185 000110 004767 000000G CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
186 000114 010067 000000G MOV R0, TEMP ; SAVE TEMPLATE ADDRESS
187 000120 012760 000000G 000000G MOV #CS.X29, C.STS(R0) ; INDICATE THAT THIS IS A X29$DF TEMPLATE
188 000126 005001 CLR R1 ; IGNORE BLANKS
189 000130 012702 000000' MOV #X29KTB, R2 ; KEYWORD TABLE
190 000134 012705 000000' MOV #X29STB, R5 ; STATE TABLE
191 000140 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
192 000144 103012 BCC 20$; NORMAL RETURN IF NO ERROR
193 000146 016700 000000G MOV ERROR, R0 ; GET PARSE ERROR
194 000152 001405 BEQ 15$; IF EQ, NONE
195 000154 ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
196 000160 012767 177776 000000G 15$: MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
197 000166 005267 000000G 20$: INC $ERROR ; INDICATE SYNTAX ERROR
198 000172 012605 MOV (SP)+, R5 ; RESTORE R5
199 000174 000207 RETURN

```

```

261 .SBTTL $TMX3P - BUILD X3P TEMPLATE
262
263 :+
264 *** - $TMX3P - BUILD X3P TEMPLATE
265 THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
266
267 INPUT:
268 R3-R5 - TPARS REGISTERS
269
270 OUTPUT:
271 C-BIT = SUCCESS/FAILURE
272 THE X3P TEMPLATE IS BUILT IN THE END OF TASK BUFFER SPACE.
273
274 -
275
276 $TMX3P::
277 MOV R5, -(SP) ; SAVE R5
278 MOV #C.LEN, R1 ; GET LENGTH OF NEEDED ALLOCATION
279 CALL $TALOC ; TRY TO ALLOCATE A CORE BLOCK
280 MOV R0, .TEMP ; SAVE TEMPLATE ADDRESS
281 MOV #CS.X3P.C.STS(R0) ; INDICATE THAT THIS IS A X3P$DF TEMPLATE
282 CLR R1 ; IGNORE BLANKS
283 MOV #X3PKTB, R2 ; KEYWORD TABLE
284 MOV #X3PSTB, R5 ; STATE TABLE
285 CALL .TPARS ; PARSE THE REST OF THE LINE
286 BCC 20$; NORMAL RETURN IF NO ERROR
287 MOV ,ERROR, R0 ; GET PARSE ERROR
288 BEQ 15$; IF EQ, NONE
289 ERROR$ R0 ; PRINT ACTION ROUTINE ERROR
290 MOV #-2, $ERROR ; SUPPRESS SYNTAX MESSAGE
291 INC $ERROR ; INDICATE SYNTAX ERROR
292 MOV (SP)+, R5 ; RESTORE R5
293 RETURN

```



UPDBUF - CFE UPDATE BUFFER DEFINITION MACRO V05.03b Saturday 29-Jun-85 05:47 Page 7  
 SETNUM - SET THE NEXT NUMBER

220  
 221  
 222  
 223  
 224  
 225  
 226  
 227  
 228  
 229  
 230  
 231  
 232  
 233  
 234  
 235  
 236  
 237  
 238  
 239  
 240  
 241  
 242  
 243  
 244

000122  
 000122 016700 000000G  
 000126 016702 177646  
 000132 012201  
 000134 010267 177640  
 000140 004767 000000G  
 000144 010067 000000G  
 000150 000207  
  
 000001

```

 .SBTTL SETNUM - SET THE NEXT NUMBER
 *** - SETNUM - SET UP THE NEXT NUMBER
 INPUT:
 $OBUF = CURRENT POSITION IN THE OUTPUT BUFFER
 NEXT = ADDRESS OF NEXT LOCATION IN THE TEMPLATE FILE
 OUTPUT:
 C-BIT = SUCCESS/FAILURE
 $OBUF = NEXT LOCATION IN OUTPUT BUFFER
 -
SETNUM:
 MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
 MOV NEXT,R2 ; GET ADDRESS OF NEXT LOCATION IN TEMPLATE
 MOV (R2)+,R1 ; GET NEXT VALUE
 MOV R2,NEXT ; SAVE NEXT LOCATION IN TEMPLATE
 CALL $CNVD ; CONVERT NUMBER TO DECIMAL
 MOV R0,$OBUF ; SAVE CURRENT POSITION IN BUFFER
 RETURN ; RETURN TO CALLER
 .END

```

UPDCHN      CREATED BY    MACRO    ON 29-JUN-85 AT 05:47      PAGE 2      K 14

MACRO CROSS REFERENCE      CREF      04.00

MACRO NAME      REFERENCES

|           |        |       |       |
|-----------|--------|-------|-------|
| EPRINT    | #4-79  |       |       |
| ERROR\$   | #4-68  |       |       |
| PRINT     | #4-63  |       |       |
| \$PTNMB   | #4-112 |       |       |
| \$PTNUM   | #4-111 | 6-155 | 6-156 |
| \$PTR50   | #4-116 |       |       |
| \$\$\$PTN | #4-91  | 4-111 | 4-112 |

UPDCNT      CREATED BY    MACRO    ON 29-JUN-85 AT 05:48      PAGE 1      K 15  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE      | REFERENCES                                              |
|-----------|------------|---------------------------------------------------------|
| CNTKTB    | 000000 RG  | #5-15 6-169                                             |
| CNTSTB    | 000000 RG  | #5-95 6-170                                             |
| CS.CNT    | = ***** GX | 6-155                                                   |
| CS.MOD    | = ***** GX | 6-162                                                   |
| C.CNT     | 000004     | #4-82 7-197                                             |
| C.CSR     | 000010     | #4-84 9-238                                             |
| C.LEN     | = 000016   | #4-87                                                   |
| C.PRI     | 000012     | #4-85 10-259                                            |
| C.STS     | = ***** GX | 6-155 6-162                                             |
| C.URM     | 000014     | #4-86 11-280                                            |
| C.VEC     | 000006     | #4-83 8-217                                             |
| F.NRBD    | = ***** GX | *6-164                                                  |
| SETCNT    | 000114 R   | #7-194                                                  |
| SETCSR    | 000170 R   | #9-236                                                  |
| SETPRI    | 000216 R   | #10-237                                                 |
| SETURM    | 000244 R   | #11-278                                                 |
| SETVEC    | 000142 R   | #8-215                                                  |
| TEMP      | 000000 R   | #4-74 *6-161 7-196 8-216 9-237 10-258 11-279            |
| \$ALPHA   | = 000022   | #5-95                                                   |
| \$ANY     | = 000020   | #5-95                                                   |
| \$BLANK   | = 000006   | #5-95                                                   |
| \$BUFI    | = ***** GX | 6-164                                                   |
| \$CNVO    | = ***** GX | 8-219 9-240 10-261 11-283                               |
| \$DIGIT   | = 000024   | #5-95                                                   |
| \$DNUMB   | = 000014   | #5-95                                                   |
| \$EOS     | = 000012   | #5-95                                                   |
| \$ERROR   | = ***** GX | *6-173                                                  |
| \$EXIT    | = 000000   | #5-95                                                   |
| \$FAIL    | = 177777   | #5-95                                                   |
| \$GPRM    | = *****    | 5-95                                                    |
| \$LAMDA   | = 000000   | #5-95                                                   |
| \$MOVPT   | = ***** GX | 7-195                                                   |
| \$MOVST   | = ***** GX | 6-166                                                   |
| \$NUMBR   | = 000002   | #5-95                                                   |
| \$OBUF    | = ***** GX | 8-218 *8-220 9-239 *9-241 10-260 *10-262 11-282 *11-284 |
| \$OFDB    | = ***** GX | *6-164                                                  |
| \$RAD50   | = 000016   | #5-95                                                   |
| \$RONLY   | = *****    | 5-95 5-95                                               |
| \$STRNG   | = 000004   | #5-95                                                   |
| \$SUBXP   | = 000010   | #5-95                                                   |
| \$TMLST   | = ***** GX | 6-151                                                   |
| \$UPCNT   | 000002 RG  | #6-150                                                  |
| \$\$\$FLG | = 177777   | #5-95                                                   |
| \$\$\$KEY | = 177777   | #5-95                                                   |
| .CONER    | = ***** GX | 6-179                                                   |
| .PNUMB    | = ***** GX | 7-197                                                   |
| .TPARS    | = ***** GX | 6-171                                                   |

UPDDDM - CFE UPDATE DEVICE PROC MACRO V05.03b Saturday 29-Jun-85 05:48 <sup>K 16</sup>  
Table of contents

|     |     |                                            |
|-----|-----|--------------------------------------------|
| 6-  | 188 | \$UPDDM - UPDATE DDM DEFINITION            |
| 7-  | 231 | SETNAM - CHECK PROCESS NAME                |
| 8-  | 255 | SETFLG - SETUP THE FLAGS WORD              |
| 9-  | 290 | SETPRI - SET THE PRIORITY                  |
| 10- | 310 | SETNLT - STORE NUMBER OF LINE TABLES       |
| 11- | 332 | SETNCT - STORE NUMBER OF CONTROLLER TABLES |

\*\*FILE\*\*ID\*\*TMPREM

```

TTTTTTTTTT MM MM PPPPPPP RRRRRRRR EEEEEEEEE MM MM
TTTTTTTTTT MM MM PPPPPPP RRRRRRRR EEEEEEEEE MM MM
 TT MMMM MMMM PP PP RR RR EE MM MM
 TT MMMM MMMM PP PP RR RR EE MM MM
 TT MM MM PP PP RR RR EE MM MM
 TT MM MM PP PP RR RR EE MM MM
 TT MM MM PPPPPPP RRRRRRRR EEEEEEEEE MM MM
 TT MM MM PPPPPPP RRRRRRRR EEEEEEEEE MM MM
 TT MM MM PP RR RR EE MM MM
 TT MM MM PP RR RR EE MM MM
 TT MM MM PP RR RR EE MM MM
 TT MM MM PP RR RR EE MM MM
 TT MM MM PP RR RR EEEEEEEEE MM MM
 TT MM MM PP RR RR EEEEEEEEE MM MM

```

```

....
....
....
....

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

11

**Abstract**

|   |   |   |   |   |   |   |   |   |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |     |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|

\*\*FILE\*\*ID\*\*TMPSER

```

TTTTTTTTTT MM MM PPPPPPP SSSSSSSS EEEEEEEEE RRRRRRRR
TTTTTTTTTT MM MM PPPPPPP SSSSSSSS EEEEEEEEE RRRRRRRR
TT MM MM PP PP SS RR RR
TT MMMM MMMM PP PP SS RR RR
TT MM MM PP PP SS RR RR
TT MM MM PP PP SS RR RR
TT MM MM PPPPPPP SSSSSS EEEEEEE RRRRRRRR
TT MM MM PPPPPPP SSSSSS EEEEEEE RRRRRRRR
TT MM MM PP SS RR RR
TT MM MM PP SS RR RR
TT MM MM PP SS RR RR
TT MM MM PP SSSSSSSS EEEEEEEEE RR RR
TT MM MM PP SSSSSSSS EEEEEEEEE RR RR

```

```

LL SSSSSSSS TTTTTTTTT?
LL SSSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LLLLLLLLLLL SSSSSSSS TT
LLLLLLLLLLL SSSSSSSS TT

```

|     |        |        |        |       |        |            |                              |
|-----|--------|--------|--------|-------|--------|------------|------------------------------|
| 453 | 000530 | 001402 |        | BEQ   | 35\$   |            | ; NO                         |
| 454 | 000532 | 010123 |        | 33\$: | MOV    | R1,(R3)+   | ; STORE CONVERTED VALUE      |
| 455 | 000534 | 103371 |        |       | BCC    | 30\$       | ; IF CC, MORE                |
| 456 | 000536 | 112614 |        | 35\$: | MOVB   | (SP)+,(R4) | ; RESTORE TRAILING CHARACTER |
| 457 | 000540 |        |        |       | RESRG  | R3         | ; RESTORE REGISTER           |
| 458 | 000542 | 000402 |        |       | BR     | 50\$       | ; THAT'S ALL, FOLKS          |
| 459 | 000544 | 062716 | 000002 | 40\$: | ADD    | #2,(SP)    | ; REJECT TRANSITION          |
| 460 | 000550 | 000207 |        | 50\$: | RETURN |            |                              |
| 461 |        |        |        |       |        |            |                              |
| 462 |        |        |        |       | .DSABL | LSB        |                              |
| 463 |        |        |        |       |        |            |                              |
| 464 |        | 000001 |        |       | .END   |            |                              |



SETNAM - SET UP PROCESS NAME

286  
287  
288  
289  
290  
291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301 000506

```
.SBTTL SETNAM - SET UP PROCESS NAME
:
: +
: *** - SETNAM - SETUP PROCESS NAME
:
: THIS ACTION ROUTINE SETS UP THE PROCESS NAME.
:
: INPUT:
: .PSTPT = START ADDRESS OF THE PROCESS NAME IN ASCII
: .PSTCN = NUMBER OF CHARACTERS IN THE PROCESS NAME
:
: OUTPUT:
: S$LNAM = RAD50 PROCESS NAME
:
: -
:
SETNAM: $GTR50 #S$LNAM,#1
```

.TITLE TMPSTA - CFE PARSE STATION DEFINITION IN CETAB  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE STATION DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
 DECNET-11M/S V3.0  
 DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0



TMPSVC      CREATED BY    MACRO    ON 29-JUN-85 AT 05:44      PAGE 3      L 8

MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|          |        |        |        |        |        |        |        |        |        |        |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| CALLR    | #4-122 |        |        |        |        |        |        |        |        |        |
| DBGTP\$  | #6-152 | #6-174 | #6-181 | #6-206 | #6-223 |        |        |        |        |        |
| EPRINT   | #4-76  |        |        |        |        |        |        |        |        |        |
| ERROR\$  | #4-65  | 7-264  |        |        |        |        |        |        |        |        |
| ISTAT\$  | #4-122 | 6-152  |        |        |        |        |        |        |        |        |
| MTRANS   | #6-152 |        |        |        |        |        |        |        |        |        |
| PRINT    | #4-60  |        |        |        |        |        |        |        |        |        |
| STATES\$ | #4-122 | 6-156  | #6-158 | #6-160 | #6-162 | #6-164 | #6-166 | #6-168 | #6-172 | #6-175 |
|          | #6-179 | #6-182 | #6-186 | #6-188 | #6-192 | #6-194 | #6-198 | #6-200 | #6-204 | #6-207 |
|          | #6-210 | #6-213 | #6-219 | #6-221 | #6-228 |        |        |        |        |        |
| SVCDF\$  | #4-122 | 4-124  |        |        |        |        |        |        |        |        |
| TRANS\$  | #4-122 | #6-157 | #6-159 | #6-161 | #6-163 | #6-165 | #6-167 | #6-169 | #6-173 | #6-174 |
|          | #6-176 | #6-180 | #6-181 | #6-183 | #6-187 | #6-189 | #6-193 | #6-195 | #6-199 | #6-201 |
|          | #6-205 | #6-206 | #6-208 | #6-211 | #6-214 | #6-220 | #6-222 | #6-223 |        |        |
| \$GTNMB  | #4-98  |        |        |        |        |        |        |        |        |        |
| \$GTNUM  | #4-88  | 8-295  | 8-297  | 8-299  | 8-301  | 8-303  | 8-305  | 8-307  |        |        |
| \$GTR50  | #4-110 | 8-309  | 8-311  |        |        |        |        |        |        |        |

```

328 .SBTTL SETPCH - SET THE PROTOCAOL EMULATOR CHARACTERISTICS
329 ;+
330 ;*** - SETPCH - SET THE PROTOCAOL EMULATOR CHARACTERISTICS
331 ;
332 ; INPUT:
333 ; .PNUMB = PROTOCAOL EMULATOR CHARACTERISTICS
334 ; .PNUMH = PROTOCAOL EMULATOR CHARACTERISTICS (HIGH ORDER)
335 ; .TEMP = TEMPLATE ADDRESS
336 ;
337 ; OUTPUT:
338 ; THE PROTOCAOL EMULATOR CHARACTERISTICS C.PCH OF THE TEMPLATE.
339 ;
340 ; -
341
342 SETPCH:
343 000344 016700 000000G MOV .TEMP,R0 ; GET START OF TEMPLATE
344 000350 016760 000000G 000024 MOV .PNUMB,C PCH(R0) ; STORE THE LINE DEAD RATIO IN THE TEMPLATE
345 000356 052760 000000G 000000G BIS #UN$PCH,C.STS(R0) ; REMEMBER OPTION
346 000364 005767 000000G TST .PNUMH ; IS IT A WORD VALUE ?
347 000370 001003 BNE 10$; IF NE, NO - REJECT
348 000372 105767 000001G TSTB .PNUMB+1 ; IS THE DEAD POLLING RATIO IN RANGE ?
349 000376 001402 BEQ 20$; IF EQ, YES
350 000400 062716 000002 10$: ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
351 000404 000207 20$: RETURN
352
353 000001 .END

```

```

269 .SBTTL SETWSZ - SET WINDOW BLOCK SIZE
270 .SBTTL SETBSZ - SET BLOCK SIZE
271 .SBTTL SETRTC - SET RETRANSMIT COUNT
272 .SBTTL SETRTT - SET RETRANSMIT TIMER
273 .SBTTL SETHBT - SET HOLDBACK TIMER
274
275 ;+
276 ** SETWSZ - SET WINDOW BLOCK SIZE
277 ** SETBSZ - SET BLOCK SIZE
278 ** SETRTC - SET RETRANSMIT COUNT
279 ** SETRTT - SET RETRANSMIT TIMER
280 ** SETHBT - SET HOLDBACK TIMER
281
282 INPUTS: .TPARS CONVENTIONS.
283
284 OUTPUT: VALUE SET IN TEMPLATE.
285
286 SETWSZ: $GTNUM #C.WSZ,#1,#7.
287
288 SETBSZ: $GTNUM #C.BSZ,#21,#1029.
289
290 SETRTC: $GTNUM #C.RTC,,#255.
291
292 SETRTT: $GTNUM #C.RTT,,#1000.
293
294 SETHBT: $GTNUM #C.HBT,,#600.
295
296
297 .SBTTL SETDEV - SET DEVICE NAME IN LINE ID
298 .SBTTL SETCON - SET CONTROLLER NUMBER IN LINE ID
299 .SBTTL SETUNI - SET UNIT NUMBER IN LINE ID
300
301 ;+
302 ** SETDEV - SET DEVICE NAME IN LINE ID
303 ** SETCON - SET CONTROLLER NUMBER IN LINE ID
304 ** SETUNI - SET UNIT NUMBER IN LINE ID
305
306 INPUTS: .TPARS CONVENTIONS.
307
308 OUTPUT: VALUE SET IN TEMPLATE.
309
310 SETDEV: $GTR50 #C.LIN,#1
311
312 SETCON: $GTNUM #C.LIN+2
313
314 SETUNI: $GTNUM #C.LIN+4,,,#CS.UNI
315
316 000001 .END

```

MPX29 - PARSE X.29 CHARACTERIS MACRO V05.03b Saturday 29-Jun-85 05:46 Page 8  
SETMXC - SET MAXIMUM CIRCUITS

```
201 .SBTTL SETMXC - SET MAXIMUM CIRCUITS
202 .SBTTL SETCT - SET CALL TIMER VALUE
203
204 +
205 ** SETMXC - SET MAXIMUM CIRCUITS
206 ** SETCT - SET CALL TIMER VALUE
207 :
208 INPUTS .TPARS VARIABLE.
209 :
210 OUTPUT VALUE PUT INTO CURRENT TEMPLATE
211 -
212 SETMXC: $GTNUM #X$9MXC
213
214 SETCT: $GTNMB #X$9CT,,#255.
215
216
217 000001 .END
```

```

295 .SBTTL SETDFP - SET DEFAULT BLOCK SIZE
296 .SBTTL SETMXP - SET MAXIMUM BLOCK SIZE
297 .SBTTL SETDW - SET DEFAULT WINDOW SIZE
298 .SBTTL SETMW - SET MAXIMUM WINDOW SIZE
299 .SBTTL SETCT - SET CALL TIMER VALUE
300 .SBTTL SETKT - SET CLEAR TIMER VALUE
301 .SBTTL SETRT - SET RESET TIMER VALUE
302 .SBTTL SETST - SET RESTART TIMER VALUE
303 .SBTTL SETKM - SET MAXIMUM CLEAR VALUE
304 .SBTTL SETRM - SET MAXIMUM RESETS VALUE
305 .SBTTL SETSM - SET MAXIMUM RESTARTS VALUE
306
307 ;+
308 ** SETDFP - SET DEFAULT BLOCK SIZE
309 ** SETMXP - SET MAXIMUM BLOCK SIZE
310 ** SETDW - SET DEFAULT WINDOW SIZE
311 ** SETMW - SET MAXIMUM WINDOW SIZE
312 ** SETCT - SET CALL TIMER VALUE
313 ** SETKT - SET CLEAR TIMER VALUE
314 ** SETRT - SET RESET TIMER VALUE
315 ** SETST - SET RESTART TIMER VALUE
316 ** SETKM - SET MAXIMUM CLEAR VALUE
317 ** SETRM - SET MAXIMUM RESETS VALUE
318 ** SETSM - SET MAXIMUM RESTARTS VALUE
319
320 INPUTS .TPARS VARIABLE.
321
322 OUTPUT VALUE PUT INTO CURRENT TEMPLATE
323
324 000704 SETDFP: $GTNUM #C.DFP,#21.,#1029.
325
326 000724 SETMXP: $GTNUM #C.MXP,#21.,#1029.
327
328 000744 SETDW: $GTNMB #C.DW,#1.,#127.
329
330 000764 SETMW: $GTNMB #C.MW,#1.,#127.
331
332 001004 SETCT: $GTNMB #C.CT.,#255.
333
334 001024 SETKT: $GTNMB #C.KT.,#255.
335
336 001044 SETRT: $GTNMB #C.RT.,#255.
337
338 001064 SETST: $GTNMB #C.ST.,#255.
339
340 001104 SETKM: $GTNMB #C.KM.,#255.
341
342 001124 SETRM: $GTNMB #C.RM.,#255.
343
344 001144 SETSM: $GTNMB #C.SM.,#255.
345
346 000001 .END
347

```



UPDBUF - CFE UPDATE BUFFER DEF1 MACRO V05.03b Saturday 29-Jun-85 05:47 Page 7-1  
 Symbol table

|                  |                   |                   |                   |                      |
|------------------|-------------------|-------------------|-------------------|----------------------|
| BUFKTB 000000RG  | 003 C.SBNM 000014 | \$ANY = 000020    | \$EXIT = 000000   | \$STRNG= 000004      |
| BUFSB 000000RG   | 002 C.SBSZ 000016 | \$BLANK= 000006   | \$FAIL = 177777   | \$SUBXP= 000010      |
| CS.BUF= ***** GX | C.STS = ***** GX  | \$BUF1 = ***** GX | \$LAMD= 000000    | \$TMLST= ***** GX    |
| CS.MOD= ***** GX | C.TSH 000020      | \$CLNUP= ***** GX | \$MOVPT= ***** GX | \$UPBUF 000002RG     |
| C.CBNM 000004    | EXIT 000064R      | \$CNVD = ***** GX | \$MOVST= ***** GX | \$\$\$\$FLG= 177777  |
| C.CBSZ 000006    | F.NRBD= ***** GX  | \$DIGIT= 000024   | \$NUMBR= 000002   | \$\$\$\$STA= 000064R |
| C.LEN = 000022   | NEXT 000000R      | \$DNUMB= 000014   | \$OBUF = ***** GX | CONER= ***** GX      |
| C.RBNM 000010    | SETNUM 000122R    | \$EOS = 000012    | \$OFDB = ***** GX | .TPARS= ***** GX     |
| C.RBSZ 000012    | \$ALPHA= 000022   | \$ERROR= ***** GX | \$RAD50= 000016   |                      |

. ABS. 000022 000 (RW,I,GBL,ABS,OVR)  
 000152 001 (RW,I,LCL,REL,CON)  
 \$STATE 000106 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000000 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000000 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10630 Words ( 42 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:24.39  
 SY:UPDBUF.V2,[132,134]UPDBUF/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDBUF

\*\*FILE\*\*ID\*\*UPDCNT

```

UU UU PPPPPPP DDDDDDDD CCCCCCCC NN NN TTTTTTTTTT
UU UU PPPPPPP DDDDDDDD CCCCCCCC NN NN TTTTTTTTTT
UU UU PP PP DD DD CC CC NN NN TT
UU UU PP PP DD DD CC CC NN NN TT
UU UU PP PP DD DD CC CC NNNN NN TT
UU UU PP PP DD DD CC CC NNNN NN TT
UU UU PPPPPPP DD DD CC CC NN NN TT
UU UU PPPPPPP DD DD CC CC NN NN TT
UU UU PP DD DD CC CC NN NN TT
UU UU PP DD DD CC CC NN NN TT
UU UU PP DD DD CC CC NN NN TT
UU UU PP DD DD CC CC NN NN TT
UU UU PP DD DD CC CC NN NN TT
UUUUUUUU PP DDDDDDDD CCCCCCCC NN NN TT
UUUUUUUU PP DDDDDDDD CCCCCCCC NN NN TT

```

```

....
....
....
....

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLL SSSSSSSS TT
LLLLLLLL SSSSSSSS TT

```

UPDCNT      CREATED BY MACRO ON 29-JUN-85 AT 05:48      PAGE 2      L 15  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-95      | #5-123 |        |        |        |        |        |        |        |        |
| ISTAT\$    | #4-68      | 5-95   |        |        |        |        |        |        |        |        |
| MIRAN\$    | #5-95      |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-59      |        |        |        |        |        |        |        |        |        |
| STATE\$    | #4-68      | 5-97   | #5-100 | #5-103 | #5-106 | #5-109 | #5-112 | #5-115 | #5-118 | #5-121 |
|            | #5-125     | #5-128 | #5-130 | #5-133 | #5-136 |        |        |        |        |        |
| TRAN\$     | #4-68      | #5-98  | #5-101 | #5-104 | #5-107 | #5-110 | #5-113 | #5-116 | #5-119 | #5-122 |
|            | #5-123     | #5-126 | #5-129 | #5-131 | #5-134 |        |        |        |        |        |

.TITLE UPDDDM - CFE UPDATE DEVICE PROCESS DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE DDM DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RX V1.0

TMPPREM - CFE PARSE REMOTE DEFIN MACRO V05.03b Saturday 29-Jun-85 <sup>N 1</sup> 05:40  
Table of contents

|     |     |                                   |      |
|-----|-----|-----------------------------------|------|
| 6-  | 127 | \$TMREM - SETUP REMOTE TEMPLATE   |      |
| 7-  | 162 | FINREM - FIND NEXT REMOTE IN SORT | LIST |
| 8-  | 195 | SETNAM - SETUP REMOTE NODE NAME   |      |
| 9-  | 226 | SETARE - SET UP REMOTE AREA       |      |
| 10- | 255 | SETADD - SET UP REMOTE ADDRESS    |      |

TMPROU - CFE PARSE ROUTING DEF I MACRO V05.03b Saturday 29-Jun-85 05:40 <sup>N 2</sup>  
Table of contents

6- 208 \$TMROU - SETUP ROUTING TEMPLATE

TMPSER - CFE PARSE REMOTE CHARA MACRO V05.03b Saturday 29-Jun-85 05:41 M 3  
Table of contents

|     |     |                                                 |
|-----|-----|-------------------------------------------------|
| 6-  | 222 | \$TMSEK - SETUP REMOTE CHARACTERISTICS TEMPLATE |
| 7-  | 259 | SETADD - SET UP REMOTE ADDRESS                  |
| 7-  | 260 | HOSADD - SET UP HOST ADDRESS                    |
| 7-  | 289 | SETARE - SET UP REMOTE AREA                     |
| 8-  | 324 | STCIR - STORE SERVICE CIRCUIT                   |
| 9-  | 352 | SERDV - STORE SERVICE DEVICE                    |
| 9-  | 353 | DADDR - STORE DUMP ADDRESS                      |
| 9-  | 354 | DCT - STORE DUMP COUNT                          |
| 10- | 396 | PAS - SERVICE BLOCK PASSWORD                    |
| 10- | 397 | STHAD - STORE HARDWARE ADDRESS                  |

|                  |                      |                    |                   |                        |
|------------------|----------------------|--------------------|-------------------|------------------------|
| ALPNUM 000350R   | 002 SERKT B 000000RG | 003 SF.DEV= 000020 | \$LEN = 000056    | \$STRNG= 000004        |
| CIRC 000266R     | 002 SERST B 000000RG | 002 SF.DPA= 000100 | \$PSS = 000034    | \$SUBXP= 000010        |
| CIRC10 000322R   | 002 SER1 000006R     | 002 SF.DPC= 000200 | S.LENX= ***** GX  | \$TALOC= ***** GX      |
| CIRC20 000334R   | 002 SER10 000060R    | 002 SF.HAD= 000004 | \$ALPHA= 000022   | \$TMSE R 000022RG      |
| CIRC30 000342R   | 002 SER15 000104R    | 002 SF.HST= 000002 | \$ANY = 000020    | \$\$\$FLG= 177777      |
| CIRC5 000306R    | 002 SER20 000126R    | 002 SF.PH3= 000400 | \$BLANK= 000006   | \$\$\$KEY= 000001      |
| CS.SER= ***** GX | SER25 000150R        | 002 SF.PSS= 000040 | \$BYTMN= ***** GX | \$\$\$STA= 000000      |
| DADDR 000344R    | SER30 000172R        | 002 STCIR 000250R  | \$CAT5 = ***** GX | \$\$\$TMP= 000007R 004 |
| DCT 000366R      | SER35 000214R        | 002 STHAD 000402R  | \$DIGIT= 000024   | .CAT5E= ***** GX       |
| ENDHEX 000022R   | SER40 000236R        | 002 S\$ADD 000006  | \$DNUMB= 000014   | .ERROR= ***** GX       |
| EXIT 000360R     | 002 SER45 000244R    | 002 S\$CIR 000022  | \$EOS = 000012    | .FLAGS= ***** GX       |
| HEXSTR 000002R   | SER5 000042R         | 002 S\$DEV 000032  | \$ERROR= ***** GX | .PNUMB= ***** GX       |
| HOSADD 000140R   | SER50 000256R        | 002 S\$DPA 000050  | \$EXIT = 000000   | .PNUMH= ***** GX       |
| HOSARE 000176R   | SETADD 000146R       | S\$DPC 000054      | \$FAIL = 177777   | .PSTCN= ***** GX       |
| PAS 000414R      | SETARE 000204R       | S\$FLG 000004      | \$LAMBDA= 000000  | .PSTPT= ***** GX       |
| REJ 000242R      | SF.ADD= 000001       | S\$HAD 000012      | \$NUMBR= 000002   | .TEMP = ***** GX       |
| SERDV 000322R    | SF.CIR= 000010       | S\$HST 000010      | \$RAD50= 000016   | .TPARS= ***** GX       |
| SERFLG 000000R   |                      |                    |                   |                        |

. ABS. 000056 000 (RW,I,GBL,ABS,OVR)  
000552 001 (RW,I,LCL,REL,CON)  
\$STATE 000364 002 (RW,D,LCL,REL,CON)  
\$KTAB 000004 003 (RW,D,LCL,REL,CON)  
\$KSTR 000016 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10452 Words ( 41 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:01:04.32  
SY:TMPSER.V2,[132,134]TMPSER/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[137,10]TMPSER



TMPSLT - CFE PARSE LINE DEFINIT MACRO V05.03b Saturday 29-Jun-85 <sup>N 5</sup> 05:42 Page 8  
 SETFLG - SETUP THE FLAGS WORD

```

303 .SBTTL SETFLG - SETUP THE FLAGS WORD
304 ;+
305 *** - SETFLG - SETUP THE FLAGS WORD
306 ;
307 INPUT:
308 FLAG = THE FLAGS WORD
309 .TEMP = START ADDRESS OF THE TEMPLATE
310 ;
311 OUTPUT:
312 THE FLAGS WORD IS MOVED INTO THE TEMPLATE
313 ;
314 -
315
316 000522 SETFLG:
317 000522 016700 000000G MOV .TEMP,R0 ; GET START OF TEMPLATE
318 000526 016760 177246 000000G MOV FLAG,$LFLG(R0) ; STORE THE FLAGS WORD
319 000534 000207 RETURN

```

```

56 ;****
57 ; LOCAL MACROS
58 ;****
59
60 .MACRO PRINT TEXT
61 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
62 CALL $PRINT ; PRINT MESSAGE
63 .ENDM PRINT
64
65 .MACRO ERROR$ TEXT
66 ;
67 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
68 ; MESSAGE STRING.
69 ;
70 .IF DIF <TEXT><RO>
71 MOV TEXT,RO ; GET ADDRESS OF ERROR MESSAGE
72 .ENDC
73 CALL $CFERR ; PRINT ERROR MESSAGE
74 .ENDM ERROR$
75
76 .MACRO EPRINT TEXT
77 ;
78 ; PRINT TEXT ON ERROR LUN
79 ;
80 .IF DIF <TEXT><RO>
81 MOV TEXT,RO ; GET ADDRESS OF ASCIZ STRING TO PRINT
82 .ENDC
83 CALL $CFERR ; PRINT MESSAGE ON ERROR LUN
84 .ENDM EPRINT
85
86 ;****
87 ; MACRO CALLS
88 ;****
89
90 .MCALL ISTAT$,STATE$,TRANS
91
92 ;****
93 ; LOCAL DATA
94 ;****
95
96 000000 .BLKW 1 ; ADDRESS OF STATION TEMPLATE
97 000002 FLAG: .BLKW 1 ; STATION FLAGS
98 .NLST BEX
99 .ENABL LC
100 EX$SEV = 4
101 000004 004 103 106 ERRSTA: .ASCIZ <EX$SEV>/CFE -- Too many stations/
102 .EVEN
103 .LIST BEX
104
105 ;
106 ; DEFINE STATION TEMPLATE
107 ;
108 000000 .ASECT
109 000000 .=0
110 000002 .BLKW 1 ; C.LNK
111 000004 C.STA: .BLKW 1 ; C.STS
112 000005 C.ADD: .BLKB 1 ; STATION NUMBER
113 ; STATION ADDRESS

```

TMPSVC - PARSE SWITCHED VIRTUAL MACRO V05.03b Saturday 29-Jun-85 <sup>N 7</sup> 05:44  
Table of contents

|    |     |                              |
|----|-----|------------------------------|
| 4- | 55  | MACRO DEFINITIONS            |
| 5- | 128 | LOCAL DATA                   |
| 6- | 147 | TPARS STATE TABLES           |
| 7- | 231 | \$TMSVC - BUILD SVC TEMPLATE |
| 8- | 271 | SETNAM - SET DLM NAME        |
| 8- | 272 | SETPSZ - SET PACKET SIZE     |
| 8- | 273 | SETWSZ - SET WINDOW SIZE     |
| 8- | 274 | SETFLG - SET FLAGS           |
| 8- | 275 | SETRCL - SET MAXIMUM RECALLS |
| 8- | 276 | SETTIM - SET RECALL TIMER    |
| 8- | 277 | SETDTE - SET DTE ADDRESS     |
| 8- | 278 | SETOWN - SET OWNER OF SVC    |

\*\*F ILE\*\*[D\*\*TMPUNT

```

TTTTTTTTTT MM MM PPPPPPP UU UU NN NN TTTTTTTTTT
TTTTTTTTTT MM MM PPPPPPP UU UU NN NN TTTTTTTTTT
 TT MMMM MMMM PP PP UU UU NN NN TT
 TT MMMM MMMM PP PP UU UU NN NN TT
 TT MM MM MM PP PP UU UU NNNN NN TT
 TT MM MM MM PP PP UU UU NNNN NN TT
 TT MM MM PPPPPPP UU UU NN NN NN TT
 TT MM MM PPPPPPP UU UU NN NN NN TT
 TT MM MM PP PP UU UU NN NNNN TT
 TT MM MM PP PP UU UU NN NNNN TT
 TT MM MM PP PP UU UU NN NN TT
 TT MM MM PP PP UU UU NN NN TT
 TT MM MM PP PP UU UU NN NN TT
 TT MM MM PP PP UU UU NN NN TT
 TT MM MM PP PP UUUUUUUUU NN NN TT
 TT MM MM PP PP UUUUUUUUU NN NN TT

```

```

....
....
....
....

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

M 9

IMPUNT - CFE PARSE UNIT DEFINIT MACRO V05.03b Saturday 29-Jun-85 05:44 Page 13-1

Symbol table

|                                      |                               |                       |                   |                   |
|--------------------------------------|-------------------------------|-----------------------|-------------------|-------------------|
| CS.UNT= ***** GX                     | LINDPR 000046R                | 002 TEMP 000000R      | \$DNUMB= 000014   | \$TALOC= ***** GX |
| C.CSR 000012                         | MXCSTL= ***** GX              | 003 UNTKTB 000000RG   | \$E05 = 000012    | \$TMUNT 000002RG  |
| C.CST 000014                         | PCHA 000064R                  | 002 UNSTB 000000RG    | \$ERROR= ***** GX | \$\$\$FLG= 177777 |
| C.CWO 000006                         | SECCSR 000024R                | 002 UN\$CST= ***** GX | \$EXIT = 000000   | \$\$\$KEY= 177777 |
| C.CW1 000010                         | SETCSR 000172R                | UN\$DPR= ***** GX     | \$FAIL = 177777   | \$\$\$STA= 000000 |
| C.DPR 000016                         | SETCST 000236R                | UN\$PCH= ***** GX     | \$LAMDA= 000000   | .FLAGS= ***** GX  |
| C.LEN = 000026                       | SETCWO 000116R                | UN\$SCS= ***** GX     | \$NUMBER= 000002  | .PNUMB= ***** GX  |
| C.PCH 000024                         | SETCW1 000144R                | \$ALPHA= 000022       | \$RAD50= 000016   | .PNUMH= ***** GX  |
| C.STS = ***** GX                     | SETDPR 000302R                | \$ANY = 000020        | \$STRNG= 000004   | .TEMP = ***** GX  |
| C.UNT 000004                         | SETPCH 000344R                | \$BLANK= 000006       | \$SUBXP= 000010   | .TPARS= ***** GX  |
| LINCST 000034R                       | 002 SETUNT 000062R            | \$DIGIT= 000024       |                   |                   |
| . ABS. 000026 000 (RW,I,GBL,ABS,OVR) |                               |                       |                   |                   |
|                                      | 000406 001 (RW,I,LCL,REL,CON) |                       |                   |                   |
| \$STATE 000102                       | 002 (RW,D,LCL,REL,CON)        |                       |                   |                   |
| \$KTAB 000000                        | 003 (RW,D,LCL,REL,CON)        |                       |                   |                   |
| \$KSTR 000000                        | 004 (RW,D,LCL,REL,CON)        |                       |                   |                   |

Errors detected: 0

### \*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10406 Words ( 41 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:28.20  
 SY:IMPUNT.V2,[132,134]IMPUNT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]IMPUNT

TMPIX2P - PARSE X.25 LEVEL 2 PRO MACRO V05.03b Saturday 29-Jun-85 05:45 Page 8-1

Symbol table

|                  |                 |                     |                       |                   |
|------------------|-----------------|---------------------|-----------------------|-------------------|
| CS.UNI= ***** GX | ERRLIN 000000R  | SETRTC 000446R      | \$DIGIT= 000024       | \$STRNG= 000004   |
| CS.X2P= ***** GX | ERRRTC 000120R  | SETRTT 000466R      | \$DNUMB= 000014       | \$SUBXP= 000010   |
| C.BSZ 000014     | ERRRTT 000162R  | SETUNI 000562R      | \$EOS = 000012        | \$TALOC= ***** GX |
| C.HBT 000022     | ERRWSZ 000030R  | SETWSZ 000406R      | \$ERROR= ***** GX     | \$TMX2P 000276RG  |
| C.LEN = 000024   | EX\$SEV= 000004 | X2PKTB 000000RG     | \$EXIT = 000000       | \$SFLG= 177777    |
| C.LIN 000004     | LINID 000126R   | 002 X2PSTB 000000RG | 003 \$FAIL = 177777   | \$SKEY= 177777    |
| C.RTC 000016     | LINO 000150R    | 002 \$ALPHA= 000022 | 002 \$GTNUM= ***** GX | \$SSTA= 000000    |
| C.RTT 000020     | SETBSZ 000426R  | \$ANY = 000020      | \$GTR50= ***** GX     | .ERROR= ***** GX  |
| C.WSZ 000012     | SETCON 000542R  | \$BIANK= 000006     | \$LAMD= 000000        | .FLAGS= ***** GX  |
| ERRBSZ 000065R   | SETDEV 000526R  | \$CrERR= ***** GX   | \$NUMBR= 000002       | .TEMP = ***** GX  |
| ERRHBT 000231R   | SETHBT 000506R  | \$CLERR= ***** GX   | \$RAD50= 000016       | .TPARS= ***** GX  |

. ABS. 000024 000 (RW,I,GBL,ABS,OVR)  
 000610 001 (RW,I,LCL,REL,CON)  
 \$STATE 000156 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000000 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000000 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 11048 Words ( 44 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:35.14  
 SY:TMPIX2P.V2,[132,134]TMPIX2P/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPIX2P

M 11

TMPX29 - PARSE X.29 CHARACTERIS MACRO V05.03b Saturday 29-Jun-85 05:46 Page 8-1  
 Symbol table

|                   |                     |                   |                   |                   |
|-------------------|---------------------|-------------------|-------------------|-------------------|
| CS.X29= ***** GX  | X\$9LEN= ***** GX   | \$CLERR= ***** GX | \$GTNUM= ***** GX | \$TMX29 000102RG  |
| C.STS = ***** GX  | X\$9MXC= ***** GX   | \$DIGIT= 000024   | \$LAMDA= 000000   | \$\$\$FLG= 177777 |
| ERRCT 000042R     | X29KTB 000000RG 003 | \$DNUMB= 000014   | \$NUMBR= 000002   | \$\$\$KEY= 177777 |
| ERRMYC 000000R    | X29STB 000000RG 002 | \$EOS = 000012    | \$RAD50= 000016   | \$\$\$STA= 000000 |
| EX\$SEV= 000004   | \$ALPHA= 000022     | \$ERROR= ***** GX | \$STRNG= 000004   | .ERROR= ***** GX  |
| SEICT 000216R     | \$ANY = 000020      | \$EXIT = 000000   | \$SUBXP= 000010   | .TEMP = ***** GX  |
| SETMXC 000176R    | \$BLANK= 000006     | \$FAIL = 177777   | \$TALOC= ***** GX | .TPARS= ***** GX  |
| X\$9CT = ***** GX | \$CFERR= ***** GX   | \$GTNMB= ***** GX |                   |                   |

|         |        |     |                    |
|---------|--------|-----|--------------------|
| . ABS.  | 000000 | 000 | (RW,I,GBL,ABS,OVR) |
|         | 000236 | 001 | (RW,I,LCL,REL,CON) |
| \$STATE | 000034 | 002 | (RW,D,LCL,REL,CON) |
| \$KTAB  | 000000 | 003 | (RW,D,LCL,REL,CON) |
| \$KSTR  | 000000 | 004 | (RW,D,LCL,REL,CON) |

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 11008 Words ( 43 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:13.02  
 SY:TMPX29 V2,[132,134]TMPX29/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPX29

|                  |                  |                 |                       |                   |
|------------------|------------------|-----------------|-----------------------|-------------------|
| CS.X3P= ***** GX | C.STS = ***** GX | SETCT 001004R   | \$ALPHA= 000022       | \$LAMDA= 000000   |
| C.CT 000012      | ERRCT 000216R    | SETDFP 000704R  | \$ANY = 000020        | \$NUMBR= 000002   |
| C.DFP 000004     | ERRDFP 000000R   | SETDW 000744R   | \$BLANK= 000006       | \$RAD50= 000016   |
| C.DW 000010      | ERRDW 000106R    | SETKM 001104R   | \$CFERR= ***** GX     | \$STRNG= 000004   |
| C.KM 000016      | ERRKM 000427R    | SETKT 001024R   | \$CLERR= ***** GX     | \$SUBXP= 000010   |
| C.KT 000013      | ERRKT 000257R    | SETMW 000764R   | \$DIGIT= 000024       | \$TALOC= ***** GX |
| C.LEN = 000021   | ERRMW 000152R    | SETMXP 000724R  | \$DNUMB= 000014       | \$TMX3P 000610RG  |
| C.MW 000011      | ERRMXP 000043R   | SETRM 001124R   | \$EOS = 000012        | \$\$\$FLG= 177777 |
| C.MXP 000006     | ERRRM 000473R    | SETRT 001044R   | \$ERRDR= ***** GX     | \$\$\$KEY= 177777 |
| C.RM 000017      | ERRRT 000321R    | SETSM 001144R   | \$EXIT = 000000       | \$\$\$STA= 000000 |
| C.RT 000014      | ERRSM 000540R    | SETST 001064R   | \$FAIL = 177777       | .ERROR= ***** GX  |
| C.SM 000020      | ERRST 000363R    | X3PKTB 000000RG | 003 \$GTNMB= ***** GX | .TEMP = ***** GX  |
| C.ST 000015      | EX\$SEV= 000004  | X3PSTB 000000RG | 002 \$GTNUM= ***** GX | .TPARS= ***** GX  |

. ABS. 000021 000 (RW,I,GBL,ABS,OVR)  
001164 001 (RW,I,LCL,REL,CON)  
\$STATE 000234 002 (RW,D,LCL,REL,CON)  
\$KTAB 000000 003 (RW,D,LCL,REL,CON)  
\$KSTR 000000 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

### \*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 11032 Words ( 44 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:45.35  
SY: TMPX3P.V2,[132,134]TMPX3P/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]TMPX3P



UPDBUF      CREATED BY MACRO ON 29-JUN-85 AT 05:47      PAGE 1      M 13  
 SYMBOL CROSS REFERENCE      CREF 04.00

| SYMBOL    | VALUE      | REFERENCES                        |
|-----------|------------|-----------------------------------|
| BUFKTB    | 000000 RG  | #5-117 6-208                      |
| BUFBTB    | 000000 RG  | #5-117 6-209                      |
| CS.BUF    | = ***** GX | 6-193                             |
| CS.MOD    | = ***** GX | 6-200                             |
| C.CBNM    | 000004     | #4-101 6-204                      |
| C.CBSZ    | 000006     | #4-102                            |
| C.LEN     | = 000022   | #4-108                            |
| C.RBNM    | 000010     | #4-103                            |
| C.RBSZ    | 000012     | #4-104                            |
| C.SBNM    | 000014     | #4-105                            |
| C.SBSZ    | 000016     | #4-106                            |
| C.STS     | = ***** GX | 6-193 6-200                       |
| C.THSH    | 000020     | #4-107                            |
| F.NRBD    | = ***** GX | *6-202                            |
| NEXT      | 000000 R   | #4-111 *6-199 *6-204 7-236 *7-238 |
| SETNUM    | 000122 R   | #7-234                            |
| \$ALPHA   | = 000022   | #5-117                            |
| \$ANY     | = 000020   | #5-117                            |
| \$BLANK   | = 000006   | #5-117                            |
| \$BUF1    | = ***** GX | 6-202                             |
| \$CNVD    | = ***** GX | 7-239                             |
| \$DIGIT   | = 000024   | #5-117                            |
| \$DNUMB   | = 000014   | #5-117                            |
| \$EOS     | = 000012   | #5-117                            |
| \$ERROR   | = ***** GX | *6-212                            |
| \$EXIT    | = 000000   | #5-117                            |
| \$FAIL    | = 177777   | #5-117                            |
| \$GPRM    | = *****    | 5-117                             |
| \$LAMDA   | = 000000   | #5-117                            |
| \$MOVST   | = ***** GX | 6-205                             |
| \$NUMBR   | = 000002   | #5-117                            |
| \$OBUF    | = ***** GX | *7-240                            |
| \$OFDB    | = ***** GX | *6-202                            |
| \$RAD50   | = 000016   | #5-117                            |
| \$RONLY   | = *****    | 5-117 5-117                       |
| \$STRNG   | = 000004   | #5-117                            |
| \$SUBYP   | = 000010   | #5-117                            |
| \$TMLST   | = ***** GX | 6-189                             |
| \$UPBUF   | 000002 RG  | #6-188                            |
| \$\$\$FLG | = 177777   | #5-117                            |
| \$\$\$KEY | = 177777   | #5-117                            |
| .CONER    | = ***** GX | 6-218                             |
| .TPARS    | = ***** GX | 6-210                             |

UPDCNT - CFE UPDATE CONTROLLER MACRO V05.03b Saturday 29-Jun-85 05:48 <sup>N 14</sup>  
Table of contents

|     |     |                                        |
|-----|-----|----------------------------------------|
| 6-  | 138 | \$UPCNT - UPDATE CONTROLLER DEFINITION |
| 7-  | 181 | SETCNT - CHECK CONTROLLER NUMBER       |
| 8-  | 202 | SETVEC - SETUP THE VECTOR ADDRESS      |
| 9-  | 223 | SETCSR - SET THE CSR ADDRESS           |
| 10- | 244 | SETPRI - SET THE PRIORITY              |
| 11- | 265 | SETURM - SET UNIBUS RUN MASK           |

```

UU UU PPPPPPP DDDDDDDD CCCCCCCC UU UU GGGGGGGG
UU UU PPPPPPP DDDDDDDD CCCCCCCC UU UU GGGGGGGG
UU UU PP PP DD DD CC CC UU UU GG
UU UU PP PP DD DD CC CC UU UU GG
UU UU PP PP DD DD CC CC UU UU GG
UU UU PP PP DD DD CC CC UU UU GG
UU UU PPPPPPP DD DD CC CC UU UU GG
UU UU PPPPPPP DD DD CC CC UU UU GG
UU UU PP DD DD CC CC UU UU GG
UU UU PP DD DD CC CC UU UU GG
UU UU PP DD DD CC CC UU UU GG
UU UU PP DD DD CC CC UU UU GG
UUUUUUUU PP DDDDDDDD CCCCCCCC UUUUUUUUU GGGGGG
UUUUUUUU PP DDDDDDDD CCCCCCCC UUUUUUUUU GGGGGG

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 .MACRO FLAG, FLG
65 .ASCII /ZF,'FLG'!/
66 Z'FLG'L = , - Z'FLG'
67 .ENDM FLAG
68
69 .MACRO PUTFLG FLA,D=<> ?LB
70 BIT #ZF,'FLA',C.FLG(R5) ; BIT SET ?
71 BEQ LB ; IF EQ, NO
72 MOV #D'Z'FLA',R1 ; ELSE POINT TO TEXT
73 MOV #D'Z'FLA'L,R2 ; GET LENGTH
74 CALL $MVASC ; MOVE IT
75
76 LB:
77 .ENDM PUTFLG
78
79 ;****
80 ; MACRO CALLS
81 ;****
82
83 .MCALL ISTAT$,STAT$,TRAN$
84
85 ;****
86 ; LOCAL DATA
87 ;****
88
89 TEMP: .BLKW 1 ; ADDRESS OF DDM TEMPLATE
90 FLAG DDM ; MARK PROCESS AS A DDM
91 FLAG DLC ; PROCESS IS A DLC
92 FLAG KMX ; PROCESS IS A KMC DEVICE
93 FLAG MUX ; PROCESS IS A MUX DEVICE
94 FLAG LMC ; PROCESS NEEDS MICRO-CODE
95 FLAG PSE ; PROCESS HAS PSEUDO SLT
96 FLAG DIA ; PROCESS HAS DIAGNOSTIC MICROCODE
97 FLAG DVP ; PROCESS RUNS AT DEVICE PRIORITY
98 FLAG MAN ; PROCESS HAS NET. MAN. ENTRY POINT
99
100 .EVEN
101
102 ; DEFINE DDM TEMPLATE
103 ;
104 .ASECT
105 .=0
106 .BLKW 1 ; C.LNK
107 .BLKW 1 ; C.STS
108 C.NAM: .BLKW 1 ; DDM PROCESS NAME
109 C.FLG: .BLKW 1 ; DDM FLAGS WORD
110 C.PRI: .BLKW 1 ; DDM PROCESS PRIORITY
111 C.NLT: .BLKB 1 ; NUMBER OF LINE TABLES IN PROCESS SPACE

```

.TITLE TMPREM - CFE PARSE REMOTE DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE REMOTE DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Mirro/RSX V1.0

.TITLE TMPCOU - CFE PARSE ROUTING DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE ROUTING DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

.TITLE TMPSER - CFE PARSE REMOTE CHARACTERISTICS IN CETAB  
.IDENT /V05.00/  
.ENABL LC

.. COPYRIGHT (C) 1983, 1985 BY  
.. DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

.. THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
.. ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
.. INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
.. COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
.. OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
.. TRANSFERRED.

.. THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
.. AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
.. CORPORATION.

.. DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
.. SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

.. MODULE DESCRIPTION:

.. STATE TABLE TO PARSE THE REMOTE CHARACTERISTICS

.. DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

.. IDENT HISTORY:

.. 4.00 07-NOV-83  
.. DECNET-11M V4.0  
.. DECNET-11M-PLUS V2.0

.. 5.00 22-JUL-85  
.. DECnet-11M/S V4.2  
.. DECnet-11M-Plus V3.0  
.. DECnet-Micro/RSX V1.0

TMPSER      CREATED BY    MACRO    ON 29-JUN-85 AT 05:42      PAGE 1      N 4  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE      | REFERENCES                    |
|---------|------------|-------------------------------|
| CS.SER  | = ***** GX | 6-243                         |
| DADDR   | 000344 R   | #9-378                        |
| DCT     | 000366 R   | #9-391                        |
| ENDHEX  | 000022 R   | #4-89                         |
| HEXSTR  | 000002 R   | #4-88                         |
| HOSADD  | 000140 R   | #7-275                        |
| HOSARE  | 000176 R   | #7-303                        |
| PAS     | 000414 R   | #10-418                       |
| REJ     | 000242 R   | 7-280      #7-318      8-337  |
| SERDV   | 000322 R   | #9-364                        |
| SERFLG  | 000000 R   | #4-87      *6-237      6-251  |
| SERKTB  | 000000 RG  | #5-96      6-247              |
| SERSTB  | 000000 RG  | #5-96      6-248              |
| SETADD  | 000146 R   | #7-277                        |
| SETARE  | 000204 R   | #7-305                        |
| SF.ADD  | = 000001   | #4-74                         |
| SF.CIR  | = 000010   | #4-77                         |
| SF.DEV  | = 000020   | #4-78                         |
| SF.DPA  | = 000100   | #4-80                         |
| SF.DPC  | = 000200   | #4-81                         |
| SF.HAD  | = 000004   | #4-76                         |
| SF.HST  | = 000002   | #4-75                         |
| SF.PH3  | = 000400   | #4-82                         |
| SF.PSS  | = 000040   | #4-79                         |
| STCIR   | 000250 R   | #8-336                        |
| STHAD   | 000402 R   | #10-414                       |
| S\$ADD  | 000006     | 7-278      7-306              |
| S\$CIR  | 000022     | #4-61      8-336      8-339   |
| S\$DEV  | 000032     | #4-64      8-336      *9-367  |
| S\$DPA  | 000050     | #4-65      *9-379      *9-380 |
| S\$DPC  | 000054     | #4-67      *9-392             |
| S\$FLG  | 000004     | #4-68      *6-251             |
| S\$HAD  | 000012     | #4-60      10-414             |
| S\$HST  | 000010     | #4-63      7-275      7-303   |
| S\$LEN  | = 000056   | #4-62      6-238              |
| S\$PSS  | 000034     | #4-69      10-418             |
| S.LENX  | = ***** GX | 6-244                         |
| \$ALPHA | = 000022   | #5-96                         |
| \$ANY   | = 000020   | #5-96                         |
| \$BLANK | = 000006   | #5-96                         |
| \$BYTMN | = ***** GX | *6-244                        |
| \$CAT5  | = ***** GX | 9-365                         |
| \$DIGIT | = 000024   | #5-96                         |
| \$DNUMB | = 000014   | #5-96                         |
| \$EOS   | = 000012   | #5-96                         |
| \$ERROR | = ***** GX | *6-253      *6-254            |
| \$EXIT  | = 000000   | #5-96                         |
| \$FAIL  | = 177777   | #5-96                         |
| \$GPRM  | = *****    | 5-96                          |
| \$LAMDA | = 000000   | #5-96                         |
| \$NUMBR | = 000002   | #5-96                         |
| \$RAD50 | = 000016   | #5-96                         |



TMPSLT - CFE PARSE LINE DEFINIT MACRO V05.03b Saturday 29-Jun-85 N 5 05:42 Page 9  
SETCNT - SET THE CONTROLLER NUMBER

321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335

000536

```
.SBTTL SETCNT - SET THE CONTROLLER NUMBER
;+
;*** - SETCNT - SET THE CONTROLLER NUMBER
;
;INPUT:
; .PNUMB = CONTROLLER NUMBER
; .PNUMH = CONTROLLER NUMBER (HIGH ORDER)
;
;OUTPUT:
; THE CONTROLLER NUMBER IS STORED IN $SLCNT OF THE TEMPLATE.
;-
SETCNT: $GTNMB #S$LCNT
```

|            |              |   |                        |
|------------|--------------|---|------------------------|
| 113 000006 | C.FLG: .BLKB | 1 | ; STATION FLAGS        |
| 114 000007 | C.CST: .BLKB | 1 | ; STATION COST         |
| 115 000010 | C.APR: .BLKB | 1 | ; ACTIVE POLLING RATIO |
| 116 000011 | .BLKB        | 1 | ; AVAILABLE            |
| 117 000012 | C.HTM: .BLKW | 1 | ; HELLO TIMER          |
| 118 000014 | .BLKW        | 1 | ; UNUSED               |
| 119 000016 | C.LEN=.      |   |                        |
| 120 000036 | .PSECT       |   |                        |

.TITLE TMPSVC - PARSE SWITCHED VIRTUAL CIRCUIT MACRO  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1984, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE SWITCHED VIRTUAL CIRCUIT DESCRIPTOR

#### DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/Rsx V1.0

IMPUNT - CFE PARSE UNIT DEFINIT MACRO V05.03b Saturday 29-Jun-85 N 8 05:44  
Table of contents

|     |     |                                                     |
|-----|-----|-----------------------------------------------------|
| 6-  | 150 | \$TMUNT - SETUP UNIT TEMPLATE                       |
| 7-  | 180 | SETUNT - SET UP UNIT NUMBER                         |
| 8-  | 206 | SETCWO - SETUP THE CHARACTERISTICS WORD 0           |
| 9-  | 228 | SETCW1 - SET THE CHARACTERISTICS WORD 1             |
| 10- | 253 | SETCSR - SET THE SECONDARY CSR ADDRESS              |
| 11- | 278 | SETCST - SET THE LINE COST                          |
| 12- | 303 | SETDPR - SET THE LINE DEAD POLLING RATIO            |
| 13- | 328 | SETPCH - SET THE PROTOCAOL EMULATOR CHARACTERISTICS |



TMPX2P      CREATED BY MACRO ON 29-JUN-85 AT 05:45      PAGE 1      N 10  
 SYMBOL CROSS REFERENCE      CREF      04.00

| SYMBOL    | VALUE      | REFERENCES                                 |
|-----------|------------|--------------------------------------------|
| CS.UNI    | = ***** GX | 8-314                                      |
| CS.X2P    | = ***** GX | 7-252                                      |
| C.BSZ     | 000014     | #5-139 8-288                               |
| C.HBT     | 000022     | #5-142 8-294                               |
| C.LEN     | = 000024   | #5-143 7-247                               |
| C.LIN     | 000004     | #5-136 *7-253 8-310 8-312 8-314            |
| C.RTC     | 000016     | #5-140 8-290                               |
| C.RTT     | 000020     | #5-141 8-292                               |
| C.WSZ     | 000012     | #5-138 8-286                               |
| ERRBSZ    | 000065 R   | #5-152                                     |
| ERRHBT    | 000231 R   | #5-155                                     |
| ERRLIN    | 000000 R   | #5-150                                     |
| ERRRTC    | 000120 R   | #5-153                                     |
| ERRRTT    | 000162 R   | #5-154                                     |
| ERRWSZ    | 000070 R   | #5-151                                     |
| EX\$SEV   | = 000004   | #5-149 5-150 5-151 5-152 5-153 5-154 5-155 |
| SETBSZ    | 000426 R   | #8-288                                     |
| SETCON    | 000542 R   | #8-312                                     |
| SETDEV    | 000526 R   | #8-310                                     |
| SETHBT    | 000506 R   | #8-294                                     |
| SETRTC    | 000446 R   | #8-290                                     |
| SETRTT    | 000466 R   | #8-292                                     |
| SETUNI    | 000562 R   | #8-314                                     |
| SETWSZ    | 000406 R   | #8-286                                     |
| X2PKTB    | 000000 RG  | 7-256                                      |
| X2PSTB    | 000000 RG  | 7-257                                      |
| \$ALPHA   | = 000022   | #6-164                                     |
| \$ANY     | = 000020   | #6-164                                     |
| \$BLANK   | = 000006   | #6-164                                     |
| \$CFERR   | = ***** GX | 7-262                                      |
| \$DIGIT   | = 000024   | #6-164                                     |
| \$DNUMB   | = 000014   | #6-164                                     |
| \$EOS     | = 000012   | #6-164                                     |
| \$ERROR   | = ***** GX | *7-263                                     |
| \$EXIT    | = 000000   | #6-164                                     |
| \$FAIL    | = 177777   | #6-164                                     |
| \$GPRM    | = *****    | 6-164                                      |
| \$GTNUM   | = ***** GX | 8-286                                      |
| \$GTR50   | = ***** GX | 8-310                                      |
| \$I.AMDA  | = 000000   | #6-164                                     |
| \$NUMBR   | = 000002   | #6-164                                     |
| \$RAD50   | = 000016   | #6-164                                     |
| \$RONLY   | = *****    | 6-164                                      |
| \$STRNG   | = 000004   | #6-164                                     |
| \$SUBXP   | = 000010   | #6-164                                     |
| \$TALOC   | = ***** GX | 7-248                                      |
| \$TMX2P   | 000276 RG  | #7-245                                     |
| \$\$\$FLG | = 177777   | #6-164                                     |
| \$\$\$KEY | = 177777   | #6-164                                     |
| .ERROR    | = ***** GX | *7-254 7-260                               |
| .FLAGS    | = ***** GX | *7-251 8-314                               |
| .TEMP     | = ***** GX | *7-249                                     |

TMPX29      CREATED BY    MACRO    ON 29-JUN-85 AT 05:46      PAGE 1      N 11  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE      | REFERENCES                   |
|-----------|------------|------------------------------|
| CS.X29    | = ***** GX | 7-187                        |
| C.STS     | = ***** GX | *7-187                       |
| ERRCT     | 000042 R   | #5-134                       |
| ERRMXC    | U00000 R   | #5-133                       |
| EX\$SEV   | = 000004 R | #5-132      5-133      5-134 |
| SECT      | 000216 R   | #8-214                       |
| SEITMXC   | 000176 R   | #8-212                       |
| X\$9CT    | = ***** GX | 8-214                        |
| X\$9LEN   | = ***** GX | 7-184                        |
| X\$9MXC   | = ***** GX | 8-212                        |
| X29KTB    | 000000 RG  | #6-143      7-189            |
| X29STB    | 000000 RG  | #6-143      7-190            |
| \$ALPHA   | = 000020   | #6-143                       |
| \$ANY     | = 000020   | #6-143                       |
| \$BLANK   | = 000006   | #6-143                       |
| \$CFERR   | = ***** GX | 7-195                        |
| \$DIGIT   | = 000024   | #6-143                       |
| \$DNUMB   | = 000014   | #6-143                       |
| \$EOS     | = 000012   | #6-143                       |
| \$ERROR   | = ***** GX | *7-197                       |
| \$EXIT    | = 000000   | #6-143                       |
| \$FAIL    | = 177777   | #6-143                       |
| \$GPRM    | = *****    | 6-143                        |
| \$GTNMB   | = ***** GX | 3-214                        |
| \$GTNUM   | = ***** GX | 8-212                        |
| \$LAMDA   | = 000000   | #6-143                       |
| \$NUMBR   | = 000002   | #6-143                       |
| \$RAD50   | = 000016   | #6-143                       |
| \$RONLY   | = *****    | 6-143      6-143             |
| \$STRNG   | = 000004   | #6-143                       |
| \$SUBXP   | = 000010   | #6-143                       |
| \$TALOC   | = ***** GX | 7-185                        |
| \$TMX29   | 000102 RG  | #7-182                       |
| \$\$\$FLG | = 177777   | #6-143                       |
| \$\$\$KEY | = 177777   | #6-143                       |
| .ERROR    | = ***** GX | 7-193                        |
| .TEMP     | = ***** GX | *7-186                       |
| .TPARS    | = ***** GX | 7-191                        |





UPDBUF      CREATED BY MACRO ON 29-JUN-85 AT 05:47      PAGE 2      N 13  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-117     | #5-162 |        |        |        |        |        |        |        |        |
| EPRINT     | #4-75      |        |        |        |        |        |        |        |        |        |
| ERROR\$    | #4-64      |        |        |        |        |        |        |        |        |        |
| ISTAT\$    | #4-88      | 5-117  |        |        |        |        |        |        |        |        |
| MTRAN\$    | #5-117     |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-59      |        |        |        |        |        |        |        |        |        |
| STATE\$    | #4-88      | 5-121  | #5-124 | #5-127 | #5-130 | #5-133 | #5-136 | #5-139 | #5-142 | #5-145 |
|            | #5-148     | #5-151 | #5-154 | #5-157 | #5-160 | #5-163 | #5-165 | #5-172 |        |        |
| TRAN\$     | #4-88      | #5-122 | #5-125 | #5-128 | #5-131 | #5-134 | #5-137 | #5-140 | #5-143 | #5-146 |
|            | #5-149     | #5-152 | #5-155 | #5-158 | #5-161 | #5-162 | #5-164 | #5-166 |        |        |

.TITLE UPDCNT - CFE UPDATE CONTROLLER DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE CONTROLLER DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

UPDCUG - UPDATE USER GROUP DEFIN MACRO V05.03b Saturday 29-Jun-85 05:48 <sup>N.15</sup>  
Table of contents

|    |     |                                            |
|----|-----|--------------------------------------------|
| 4- | 57  | MACRO CALLS AND LOCAL DATA                 |
| 4- | 58  | MACRO DEFINITIONS                          |
| 5- | 131 | LOCAL DATA                                 |
| 6- | 152 | \$UPCUG - UPDATE PERMANENT VIRTUAL CIRCUIT |



000013  
114 000102 000014

C.NCT: .BLKB 1  
C.LEN=. .PSECT

; NUMBER OF CONTROLLER TABLES

```

UU UU PPPPPPP DDDDDDDD DDDDDDDD EEEEEEEEE CCCCCCCC
UU UU PPPPPPP DDDDDDDD DDDDDDDD EEEEEEEEE CCCCCCCC
UU UU PP PP DD DD DD DD EE CC
UU UU PP PP DD DD DD DD EE CC
UU UU PP PP DD DD DD DD EE CC
UU UU PP PP DD DD DD DD EE CC
UU UU PPPPPPP DD DD DD DD EEEEEEE CC
UU UU PPPPPPP DD DD DD DD EEEEEEE CC
UU UU PP DD DD DD DD EE CC
UU UU PP DD DD DD DD EE CC
UU UU PP DD DD DD DD EE CC
UU UU PP DD DD DD DD EE CC
UUUUUUUU PP DDDDDDDD DDDDDDDD EEEEEEEEE CCCCCCCC
UUUUUUUU PP DD DD DDDDDDDD EEEEEEEEE CCCCCCCC

```

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL S TT
LLLLLLLL SSSSSSS TT
LLLLLLLL SSSSSSS TT

```

```

306
307
308
309
310
311
312
313
314
315
316
317
318 000330
319 000330 016702 177444
320 000334 005001
321 000336 156201 000017
322 000342 016700 000000G
323 000346 004767 000000G
324 000352 010067 000000G
325 000356 000207

.SBTTL SETDEW - DELAY WEIGHT
;+
;*** - SETDEW - SETUP DELAY WEIGHT
;
;INPUT:
; TEMP = TEMPLATE ADDRESS
; $OBUF = CURRENT POSITION IN OUTPUT BUFFER
;
;OUTPUT:
;-
SETDEW:
MOV TEMP,R2 ; GET START OF TEMPLATE
CLR R1 ; GET DELAY WEIGHT
BISB C,DELW(R2),R1 ;
MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
CALL $CNVD ; CONVERT NUMBER TO DECIMAL
MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
RETURN

```

```

96 ;****
97 ; INITIALIZE TABLE GENERATION
98 ;****
99
100 000020 ISTAT$ DLCSTB,DLCKTB
101
102 000020 STATES$
103 000020 TRANS$ $RAD50,,SETNAM
104
105 000020 STATES$
106 000020 TRANS$ <','>,, $MOVPT
107
108 000020 STATES$
109 000020 TRANS$!ZF
110
111 000020 STATES$
112 000020 TRANS$ "DLC"
113
114 000020 STATES$ BITS
115 000020 TRANS$ '! ,BITO
116 000020 TRANS$ $LAMDA,PRIOR,SETFLG
117
118 000020 STATES$ BITO
119 000020 TRANS$!ZF
120
121 000020 STATES$
122 000020 TRANS$ 'TIM',BITS
123 000020 TRANS$ 'MFL',BITS
124 000020 TRANS$ 'COU',BITS
125 000020 TRANS$ 'MAN',BITS
126
127 000020 STATES$ PRIOR
128 000020 TRANS$ <','>,, $MOVPT
129
130 000020 STATES$
131 000020 TRANS$ $NUMBR,,SETPRI
132
133 000020 STATES$
134 000020 TRANS$ $EOS,EXIT
135 000020 TRANS$ <','>,, $MOVPT
136
137 000020 STATES$
138 000020 TRANS$ $NUMBR,NCT,SETNLT
139 000020 TRANS$ $LAMDA
140
141 000020 STATES$ NCT
142 000020 TRANS$ $EOS,EXIT
143 000020 TRANS$ <','>,, $MOVPT
144
145 000020 STATES$
146 000020 TRANS$ $NUMBR,EXIT,SETNCT ; NUMBER OF CONTROLLER TABLES
147
148 000020 STATES$ ZF
149 000020 TRANS$ 'ZF'
150
151 000020 STATES$
152 000020 TRANS$ '., $EXIT

```



UPDDSA - UPDATE DESTINATION ADD MACRO V05.03b Saturday 29-Jun-85 05:50 <sup>B 5</sup>  
Table of contents

|    |     |                                            |
|----|-----|--------------------------------------------|
| 4- | 57  | MACRO CALLS AND LOCAL DATA                 |
| 4- | 58  | MACRO DEFINITIONS                          |
| 5- | 131 | LOCAL DATA                                 |
| 6- | 153 | \$UPDSA - UPDATE PERMANENT VIRTUAL CIRCUIT |

```

114 ; PUT RAD50 TEMPLATE FIELD TO BUFFER
115 ;
116 .MACRO $PTR50 OFS,WC,OPT,?L
117 .IF NB <OPT>
118 MOV OPT,R2
119 CALL $PTOPT
120 BCS L
121 .IFTF
122 MOV OFS,R2
123 MOV WC,R1
124 CALL $PTR50
125 .IFT
126 L:
127 .ENDC
128 .ENDM
129

```

UPDDST - UPDATE DESTINATION DES MACRO V05.03b Saturday 29-Jun-85 05:51 Page 6-1  
Symbol table

|                 |                  |                   |                   |                  |
|-----------------|------------------|-------------------|-------------------|------------------|
| C.TSK= ***** GX | C.TSK 000014     | \$BUFI = ***** GX | \$PIOPT= ***** GX | .BUFF = ***** GX |
| C.LEN = 000020  | C.TYP 000004     | \$MVASC= ***** GX | \$PTR50= ***** GX | .COMMA= ***** GX |
| C.NAM 000006    | DSTDF 000000R    | \$OBUF = ***** GX | \$UPDST 000010RG  | .FLAGS= ***** GX |
| C.OBJ 000013    | DSTLN = 000010   | \$OFDB = ***** GX | \$WDATA= ***** GX | .TEMP = ***** GX |
| C.PRI 000012    | F.NRBD= ***** GX | \$PTNMB= ***** GX |                   |                  |

. ABS 000020 000 (RW,I,GBL,ABS,OVR)  
000214 001 (RW,I,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 8830 Words ( 35 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:05.38  
SY:UPDDST.V2,[132,134]UPDDST/CR/-SP=SY:[1,1]RSXMCM.FML/ML,[130,110]NETLIB/ML,[132,10]UPDDST

UPDDTE      CREATED BY    MACRO    ON 29-JUN-85 AT 05:51      PAGE 2      B 8

MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|           |        |       |       |       |       |       |
|-----------|--------|-------|-------|-------|-------|-------|
| CALLR     | #4-132 |       |       |       |       |       |
| EPRINT    | #4-79  |       |       |       |       |       |
| ERROR\$   | #4-68  |       |       |       |       |       |
| PRINT     | #4-63  |       |       |       |       |       |
| \$PTNMB   | #4-112 |       |       |       |       |       |
| \$PTNUM   | #4-111 | 6-190 | 6-191 | 7-230 | 7-236 | 7-242 |
| \$PTR50   | #4-116 | 6-188 | 6-193 | 7-226 |       |       |
| \$\$\$PTN | #4-91  | 4-111 | 4-112 |       |       |       |

UPDEVT      CREATED BY    MACRO    ON 29-JUN-85 AT 05:51      PAGE 1      B 9

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL | VALUE      | REFERENCES                      |
|--------|------------|---------------------------------|
| BUFF   | 000062 R   | #5-95 7-170 7-183 7-198         |
| CONL   | = 000007   | #5-94 11-292                    |
| CONSLE | = 000053 R | #5-93 5-94 11-291               |
| CS.EVT | = ***** GX | 7-176                           |
| C.CLS  | 000004     | #5-104 8-222                    |
| C.EVT  | 000006     | #5-105 8-221                    |
| C.LEN  | = 000020   | #5-107                          |
| C.SNK  | 000016     | #5-106 11-281                   |
| C.STS  | = ***** GX | 7-176                           |
| EVTDEF | 000004 R   | #5-85 5-86 7-191                |
| EVTDF  | 000030 R   | #5-87 5-88 7-184                |
| EVTKTG | 000000 RG  | #6-118 7-189                    |
| EVTLEN | = 000023   | #5-86 7-190                     |
| EVTLN  | = 000010   | #5-88 7-185                     |
| EVTOFF | 000002 R   | #5-84 *8-221 9-241 *9-243       |
| EVTSTB | 000000 RG  | #6-118 7-192                    |
| FF.ADD | = 000020   | #4-73 4-73                      |
| FF.CIR | = 000040   | #4-73 4-73                      |
| FF.CON | = 000001   | #4-73                           |
| FF.FIL | = 000002   | #4-73 11-289                    |
| FF.HST | = 040000   | #4-73                           |
| FF.LIN | = 000010   | #4-73 4-73                      |
| FF.MOD | = 000100   | #4-73 4-73                      |
| FF.MON | = 000004   | #4-73 11-285                    |
| FF.MSK | = 000077   | #4-73                           |
| FF.PRT | = 000200   | #4-73 4-73                      |
| FF.QLL | = 000370   | #4-73                           |
| FF.REM | = 100000   | #4-73                           |
| FILE   | 000047 R   | #5-91 5-92 11-287               |
| FILEL  | = 000004   | #5-92 11-288                    |
| F.ADD  | 000016     | #4-73                           |
| F.CEV  | 000030     | #4-73                           |
| F.CIR  | 000024     | #4-73                           |
| F.CLS  | 000002     | #4-73                           |
| F.EVT  | 000004     | #4-73                           |
| F.FLG  | 000014     | #4-73                           |
| F.LEN  | 000016     | #4-73                           |
| F.LIN  | 000020     | #4-73                           |
| F.LNK  | 000000     | #4-73                           |
| F.LND  | 000022     | #4-73                           |
| F.NRBD | = ***** GX | *7-170 *7-199 *7-203            |
| F.REM  | 000026     | #4-73                           |
| F.SEV  | 000004     | #4-73                           |
| LA     | = 000074   | #5-81                           |
| MONITR | 000040 R   | #5-89 5-90 11-283               |
| MONL   | = 000007   | #5-90 11-284                    |
| RA     | = 000076   | #5-82                           |
| SETCLS | 000360 R   | #8-220                          |
| SETMSK | 000400 R   | #9-240                          |
| SETNUM | 000422 R   | 8-223 9-244 #10-258             |
| SETSNK | 000440 R   | #11-279                         |
| TEMP   | 000000 R   | #5-83 *7-182 8-220 9-240 11-280 |

UPDEVT      CREATED BY    MACRO    ON 29-JUN-85 AT 05:51      PAGE 2      C 9

SYMBOL CROSS REFERENCE      CREF    04.00

UPDFEA - CFE UPDATE FEATURES DE MACRO V05.03b Saturday 29-Jun-85 05:52 <sup>B 10</sup>  
Table of contents

6- 126 \$UPFEA - UPDATE FEATURES DEFINITION  
7- 170 SETWDS - UPDATE THE CURRENT STATES WORD

|    |        |     |     |     |                         |                                       |
|----|--------|-----|-----|-----|-------------------------|---------------------------------------|
| 83 | 000007 |     |     |     | ZX3PL=-ZX3P             |                                       |
| 84 | 000036 | 132 | 106 | 056 | ZSLI: .ASCII /ZF.SLI!// | ; PROCESS USES SYSTEM LEVEL INTERFACE |
|    | 000041 | 123 | 114 | 111 |                         |                                       |
|    | 000044 | 041 |     |     |                         |                                       |
| 85 | 000007 |     |     |     | ZSLIL=-ZSLI             |                                       |
| 86 | 000045 | 132 | 106 | 056 | ZINI: .ASCII /ZF.INI!// | ; PROCESS REQUIRES INIT/EXIT SUPPORT  |
|    | 000050 | 111 | 116 | 111 |                         |                                       |
|    | 000053 | 041 |     |     |                         |                                       |
| 87 | 000007 |     |     |     | ZINIL=-ZINI             |                                       |
| 88 |        |     |     |     | .EVEN                   |                                       |

UPDLLC      CREATED BY    MACRO    ON 29-JUN-85 AT 05:52      PAGE 2      B 12

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE       | REFERENCES |        |         |        |        |         |        |         |        |
|-----------|-------------|------------|--------|---------|--------|--------|---------|--------|---------|--------|
| \$DNUMB   | = 000014    | #5-94      |        |         |        |        |         |        |         |        |
| \$EOS     | = 000012    | #5-94      |        |         |        |        |         |        |         |        |
| \$ERROR   | = ***** GX  | *6-218     |        |         |        |        |         |        |         |        |
| \$EXIT    | = 000000    | #5-94      |        |         |        |        |         |        |         |        |
| \$FAIL    | = 177777    | #5-94      |        |         |        |        |         |        |         |        |
| \$GPRM    | = *****     | 5-94       |        |         |        |        |         |        |         |        |
| \$LAMDA   | = 000000    | #5-94      |        |         |        |        |         |        |         |        |
| \$MOVPT   | = ***** GX  | 7-240      |        |         |        |        |         |        |         |        |
| \$MOVST   | = ***** GX  | 6-211      |        |         |        |        |         |        |         |        |
| \$MVASC   | = ***** GX  | 8-267      | 8-272  | 8-277   | 8-282  | 8-287  | 8-292   | 8-297  | 8-302   |        |
| \$NUMBR   | = 000002    | #5-94      |        |         |        |        |         |        |         |        |
| \$OBUF    | = ***** GX  | 8-264      | *8-304 | 9-324   | *9-326 | 10-345 | *10-347 | 11-366 | *11-368 | 12-386 |
|           |             | *12-388    | 13-406 | *13-408 |        |        |         |        |         |        |
| \$OFDB    | = ***** GX  | *6-209     |        |         |        |        |         |        |         |        |
| \$RAD50   | = 000016    | #5-94      |        |         |        |        |         |        |         |        |
| \$RONLY   | = *****     | 5-94       | 5-94   | 5-94    |        |        |         |        |         |        |
| \$STRNG   | = 000004    | #5-94      |        |         |        |        |         |        |         |        |
| \$SUBXP   | = 000010    | #5-94      |        |         |        |        |         |        |         |        |
| \$TMLST   | = ***** GX  | 6-196      |        |         |        |        |         |        |         |        |
| \$UPLLC   | = 000054 RG | #6-195     |        |         |        |        |         |        |         |        |
| \$ZMFL    | = ***** GX  | 8-275      |        |         |        |        |         |        |         |        |
| \$ZMFL    | = ***** GX  | 8-276      |        |         |        |        |         |        |         |        |
| \$ZTIM    | = ***** GX  | 8-270      |        |         |        |        |         |        |         |        |
| \$ZTIML   | = ***** GX  | 8-271      |        |         |        |        |         |        |         |        |
| \$\$\$FLG | = 177777    | #5-94      |        |         |        |        |         |        |         |        |
| \$\$\$KEY | = 177777    | #5-94      |        |         |        |        |         |        |         |        |
| .CONER    | = ***** GX  | 6-224      |        |         |        |        |         |        |         |        |
| .PSTPT    | = ***** GX  | 7-241      |        |         |        |        |         |        |         |        |
| .TPARS    | = ***** GX  | 6-216      |        |         |        |        |         |        |         |        |



\*\*FILE\*\*ID\*\*UPDNOD

```

UU UU PPPPPPP DDDDDDD NN NN 000000 DDDDDDD
UU UU PPPPPPP DDDDDDD NN NN 000000 DDDDDDD
UU UU PP PP DD DD NN NN 00 DD DD
UU UU PP PP DD DD NN NN 00 DD DD
UU UU PP PP DD DD NN NN 00 DD DD
UU UU PP PP DD DD NN NN 00 DD DD
UU UU PPPPPPP DD DD NN NN 00 DD DD
UU UU PPPPPPP DD DD NN NN 00 DD DD
UU UU PP DD DD NN NN 00 DD DD
UU UU PP DD DD NN NN 00 DD DD
UU UU PP DD DD NN NN 00 DD DD
UU UU PP DD DD NN NN 00 DD DD
UUUUUUUU PP DDDDDDD NN NN 000000 DDDDDDD
UUUUUUUU PP DDDDDDD NN NN 000000 DDDDDDD

```

....  
....  
....  
....

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LLLLLLLL SSSSSSS TT
LLLLLLLL SSSSSSS TT

```

UPDNOD      CREATED BY    MACRO    ON 29-JUN-85 AT 05:53      PAGE 2      B 14

MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |  |  |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--|--|
| DBGTP\$    | #5-101     | #5-127 | #5-140 | #5-155 |        |        |        |        |        |        |  |  |
| ISTAT\$    | #4-69      | 5-101  |        |        |        |        |        |        |        |        |  |  |
| MTRAN\$    | #5-101     |        |        |        |        |        |        |        |        |        |  |  |
| PRINT      | #4-60      |        |        |        |        |        |        |        |        |        |  |  |
| STAT\$     | #4-69      | 5-104  | #5-107 | #5-110 | #5-113 | #5-116 | #5-119 | #5-122 | #5-125 | #5-129 |  |  |
|            | #5-132     | #5-135 | #5-138 | #5-142 | #5-145 | #5-150 | #5-153 | #5-157 |        |        |  |  |
| TRAN\$     | #4-69      | #5-105 | #5-108 | #5-111 | #5-114 | #5-117 | #5-120 | #5-123 | #5-126 | #5-127 |  |  |
|            | #5-130     | #5-133 | #5-136 | #5-139 | #5-140 | #5-143 | #5-146 | #5-151 | #5-154 | #5-155 |  |  |

UPDOBJ      CREATED BY    MACRO    ON 29-JUN-85 AT 05:54      PAGE 1      B 15

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE       | REFERENCES                                                 |
|-----------|-------------|------------------------------------------------------------|
| BUFF      | 000027 R    | #4-81 6-154 6-167 6-182                                    |
| CS.OBJ    | = ***** GX  | 6-160                                                      |
| C.CPY     | 000012      | #4-93 11-315                                               |
| C.FLG     | 000005      | #4-91 9-266 10-291                                         |
| C.LEN     | = 000013    | #4-95                                                      |
| C.NAM     | 000006      | #4-92 8-231 8-233 8-238 8-241                              |
| C.STS     | = ***** GX  | 6-160                                                      |
| C.TYP     | 000004      | #4-90 7-209                                                |
| F.NRBD    | = ***** GX  | *6-154 *6-183 *6-187                                       |
| LA        | = 000074    | #4-74                                                      |
| OBJDEF    | 000002 R    | #4-77 4-78 6-175                                           |
| OBJDF     | 000017 R    | #4-79 4-80 6-168                                           |
| OBJKTB    | = 000000 RG | #5-104                                                     |
| OBJLEN    | = 000014    | #4-78 6-174                                                |
| OBJLN     | = 000010    | #4-80 6-169                                                |
| OBJSTB    | = 000000 RG | #5-104 6-176                                               |
| RA        | = 000076    | #4-75                                                      |
| SETCPY    | 000534 R    | #11-312                                                    |
| SETFLG    | 000500 R    | #10-287                                                    |
| SETNAM    | 000356 R    | #8-229                                                     |
| SETVfy    | 000444 R    | #9-262                                                     |
| SETYPE    | 000326 R    | #7-205                                                     |
| TEMP      | 000000 R    | #4-76 *6-166 7-207 8-230 8-240 9-264 10-289 11-314         |
| \$ALPHA   | = 000022    | #5-104                                                     |
| \$ANY     | = 000020    | #5-104                                                     |
| \$BLANK   | = 000006    | #5-104                                                     |
| \$BUFI    | = ***** GX  | 6-187                                                      |
| \$CNVD    | = ***** GX  | 7-210                                                      |
| \$CNVO    | = ***** GX  | 9-268 10-293 11-316                                        |
| \$CSTA    | = ***** GX  | 8-239                                                      |
| \$DIGIT   | = 000024    | #5-104                                                     |
| \$DNUMB   | = 000014    | #5-104                                                     |
| \$EOS     | = 000012    | #5-104                                                     |
| \$ERROR   | = ***** GX  | *6-179                                                     |
| \$EXIT    | = 000000    | #5-104                                                     |
| \$FAIL    | = 177777    | #5-104                                                     |
| \$GPRM    | = ***-***   | 5-104                                                      |
| \$LAMDA   | = 000000    | #5-104                                                     |
| \$MVASC   | = ***** GX  | 6-170                                                      |
| \$NUMBR   | = 000002    | #5-104                                                     |
| \$OBUF    | = ***** GX  | *6-171 6-181 7-206 *7-211 8-236 *8-244 9-263 *9-269 10-288 |
|           |             | *10-294 11-313 *11-317                                     |
| \$OFDB    | = ***** GX  | *6-154 *6-183 *6-187                                       |
| \$RAD50   | = 000016    | #5-104                                                     |
| \$RONLY   | = *****     | 5-104 5-104                                                |
| \$STRNG   | = 000004    | #5-104                                                     |
| \$SUBXP   | = 000010    | #5-104                                                     |
| \$TMLST   | = ***** GX  | 6-156                                                      |
| \$UPDOBJ  | = 000150 RG | #6-153                                                     |
| \$WDATA   | = ***** GX  | 6-184                                                      |
| \$\$\$FLG | = 177777    | #5-104                                                     |
| \$\$\$KEY | = 177777    | #5-104                                                     |

```

275 .SBTTL SETUP - SETUP TOP FLAG
276 ;
277 *** - SETUP - SETUP TOP FLAG
278 ;
279 INPUT:
280 (TEMP) = TEMPLATE ADDRESS
281 (TEMP)+C.STS = FLAGS WORD
282 ;
283 OUTPUT:
284 ;
285 ;
286 ;
287 SETUP:
288 000324 016700 177450 MOV TEMP,R0 ; POINT AT TEMPLATE
289 000330 032760 000000G 000000G BIT #CS.TOP,C.STS(R0) ; TOP DOWN LOADING ?
290 000336 001412 BEQ 10$; IF EQ, NO ... LEAVE
291 000340 016700 000000G MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
292 000344 012701 000002' MOV #TOPSW,R1 ; POINT R1 AT INPUT STRING
293 000350 012702 000003' MOV #TOPL,R2 ; AND LENGTH
294 000354 004767 000000G CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
295 000360 010067 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
296 000364 000207 10$: RETURN
297

```

```

116
117 ;****
118 ; INITIALIZE TABLE GENERATION
119 ;****
120 000102 ISTAT$ DDMSTB,DDMKTB
121
122 000102 STATES$
123 000102 TRANS$ $RAD50,,SETNAM
124
125 000102 STATES$
126 000102 TRANS$ <','>,, $MOVPT
127
128 000102 STATES$
129 000102 TRANS$!ZF
130
131 000102 STATES$
132 000102 TRANS$ "DDM"
133
134 000102 STATES$ BITS
135 000102 TRANS$ '! ,BITO
136 000102 TRANS$ $LAMBDA,PRIOR,SETFLG
137
138 000102 STATES$ BITO
139 000102 TRANS$!ZF
140
141 000102 STATES$
142 000102 TRANS$ 'TIM',BITS
143 000102 TRANS$ 'MFL',BITS
144 000102 TRANS$ 'DLC',BITS
145 000102 TRANS$ 'MUX',BITS
146 000102 TRANS$ 'LMC',BITS
147 000102 TRANS$ 'KMX',BITS
148 000102 TRANS$ 'COU',BITS
149 000102 TRANS$ 'PSE',BITS
150 000102 TRANS$ 'DIA',BITS
151 000102 TRANS$ 'DVP',BITS
152 000102 TRANS$ 'MAN',BITS
153
154 000102 STATES$ PRIOR
155 000102 TRANS$ <','>,, $MOVPT
156
157 000102 STATES$
158 000102 TRANS$ $NUMBR,,SETPRI
159
160 000102 STATES$
161 000102 TRANS$ $EOS,EXIT
162 000102 TRANS$ <','>,, $MOVPT
163
164 000102 STATES$
165 000102 TRANS$ $NUMBR,NCT,SETNLT
166 000102 TRANS$ $LAMBDA
167
168 000102 STATES$ NLT
169 000102 TRANS$ $EOS,EXIT
170 000102 TRANS$ <','>,, $MOVPT
171
172 000102 STATES$; NUMBER OF CONTROLLER TABLES

```

|     |     |                                                         |
|-----|-----|---------------------------------------------------------|
| 4-  | 44  | MACRO DEFINITIONS                                       |
| 5-  | 61  | DEC\$DF PARSE TABLES                                    |
| 6-  | 139 | \$UPDEC - UPDATE DEC DEFINITION                         |
| 7-  | 182 | SETLKS - NUMBER OF LOGICAL LINKS PROCESS SHOULD SUPPORT |
| 8-  | 202 | SETNDC - NUMBER OF NODE COUNTERS                        |
| 9-  | 222 | SETINC - INCOMING TIMER                                 |
| 10- | 243 | SETCUT - OUTGOING TIMER                                 |
| 11- | 264 | SETINA - INACTIVITY TIMER                               |
| 12- | 285 | SETDEF - DELAY FACTOR                                   |
| 13- | 306 | SETDEW - DELAY WEIGHT                                   |
| 14- | 327 | SETIPL - INPUT PACKET LIMITER                           |
| 15- | 348 | SETLST - LINK SERVICE THRESHOLD                         |
| 16- | 369 | SETRET - RETRANSMIT FACTOR                              |
| 17- | 390 | SETSEG - ECL SEGMENT SIZE                               |

```

327
328
329
330
331
332
333
334
335
336
337
338
339 000360
340 000360 016702 177414
341 000364 005001
342 000366 156201 000020
343 000372 016700 000000G
344 000376 004767 000000G
345 000402 010067 000000G
346 000406 000207

 .SBTTL SETIPL - INPUT PACKET LIMITER
 ;+
 *** - SETIPL - SETUP INPUT PACKET LIMITER
 :
 INPUT:
 : TEMP = TEMPLATE ADDRESS
 : $OBUF = CURRENT POSITION IN OUTPUT BUFFER
 :
 : OUTPUT:
 :
 :-
 SETIPL:
 MOV TEMP,R2 ; GET START OF TEMPLATE
 CLR R1 ; GET INPUT PACKET LIMITER
 BISB C,IPL(R2),R1 ;
 MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
 CALL $CNVD ; CONVERT NUMBER TO DECIMAL
 MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
 RETURN

```

```
153
154
155
156 000020
157 000020
158
159 000020
```

```
;
; EXIT STATE
;
STATE$ EXIT
TRANS $EOS,$EXIT,$CLNUP
STATE$
```



1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55

.TITLE UPDDSA - UPDATE DESTINATION ADDRESS IN CETAB  
.IDENT /V05.00/  
.ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

UPDATE DESTINATION ADDRESS IN CETAB

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-AUG-81  
DECNET-11M/S V3.1  
DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

LOCAL DATA

```

131 .SBTTL LOCAL DATA
132
133 :
134 : DEFINE THE DSC DESCRIPTOR BLOCK
135 :
136 000000 .ASECT
137 :=0
138 000000 .BLKW 1 ; C.LNK
139 000002 .BLKW 1 ; C.STS
140 000004 C.MSK: .BLKW 11. ; CALL MASK (1.-32. HEX DIGITS)
141 000032 C.VAL: .BLKW 11. ; CALL VALUE (1.-32. HEX DIGITS)
142 C.LEN=. ; LENGTH OF THE BLOCK
143 000000
144
145 000000 011 104 123 DSCDF: .NLIST BEX
146 000010 .ASCII: <11>/DSC$DF/<11>
147 DSCLN=. -DSCDF
148 .LIST BEX
149 .EVEN

```

UPDDST CREATED BY MACRO ON 29-JUN-85 AT 05:51 PAGE 1 C 7  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL  | VALUE       | REFERENCES           |
|---------|-------------|----------------------|
| CS.TSK  | = ***** GX  | 6-181                |
| C.LEN   | = 000020    | #5-145               |
| C.NAM   | 000006      | #5-141 6-178         |
| C.OBJ   | 000013      | #5-143 6-180         |
| C.PRI   | 000012      | #5-142 6-179         |
| C.TSK   | 000014      | #5-144 6-181         |
| C.TYP   | 000004      | #5-140 6-177         |
| DSTDF   | 000000 R    | #5-148 5-149 6-173   |
| DSTLN   | = 000010    | #5-149 6-174         |
| F.NRBD  | = ***** GX  | *6-167 *6-184 *6-186 |
| \$BUF1  | = ***** GX  | 6-186                |
| \$MVASC | = ***** GX  | 6-175                |
| \$OBUF  | = ***** GX  | *6-176 6-182         |
| \$CFDB  | = ***** GX  | *6-167 *6-184 *6-186 |
| \$PTNMB | = ***** GX  | 6-179 6-180          |
| \$PTOPT | = ***** GX  | 6-181                |
| \$PTR50 | = ***** GX  | 6-177 6-178 6-181    |
| \$UPDST | = 000010 RG | #6-166               |
| \$WDATA | = ***** GX  | 6-185                |
| .BUFF   | = ***** GX  | 6-167 6-172 6-183    |
| .COMMA  | = ***** GX  | *6-171               |
| .FLAGS  | = ***** GX  | *6-170               |
| .TEMP   | = ***** GX  | *6-168               |

\*\*FILE\*\*ID\*\*UPDEVT

```

UU UU PPPPPPP DDDDDDDD EEEEEEEEE VV VV TTTTTTTTTT
UU UU PPPPPPP DDDDDDDD EEEEEEEEE VV VV TTTTTTTTTT
UU UU PP PP DD DD EE VV VV TT
UU UU PP PP DD DD EE VV VV TT
UU UU PP PP DD DD EE VV VV TT
UU UU PP PP DD DD EE VV VV TT
UU UU PPPPPPP DD DD EEEEEEEEE VV VV TT
UU UU PPPPPPP DD DD EEEEEEEEE VV VV TT
UU UU PP DD DD EE VV VV TT
UU UU PP DD DD EE VV VV TT
UU UU PP DD DD EE VV VV TT
UU UU PP DD DD EE VV VV TT
UUUUUUUUUU PP DDDDDDDD EEEEEEEEE VV VV TT
UUUUUUUUUU PP DDDDDDDD EEEEEEEEE VV VV TT

```

```

LL SSSSSSS TTTTTTTTTT
LL SSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSS TT
LL SSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSSS TT
LLLLLLLLLLLL SSSSSSS TT
LLLLLLLLLLLL SSSSSSS TT

```

UPDEVT      CREATED BY    MACRO    ON 29-JUN-85 AT 05:51      PAGE 2      C 9

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE       | REFERENCES                                                          |
|-----------|-------------|---------------------------------------------------------------------|
| \$ALPHA   | = 000022    | #6-118                                                              |
| \$ANY     | = 000020    | #6-118                                                              |
| \$BLANK   | = 000006    | #6-118                                                              |
| \$BUFI    | = ***** GX  | 7-203                                                               |
| \$CNVO    | = ***** GX  | 10-260                                                              |
| \$DIGIT   | = 000024    | #6-118                                                              |
| \$DNUMB   | = 000014    | #6-118                                                              |
| \$FOS     | = 000012    | #6-118                                                              |
| \$ERROR   | = ***** GX  | *7-195                                                              |
| \$EXIT    | = 000000    | #6-118                                                              |
| \$FAIL    | = 177777    | #6-118                                                              |
| \$GPRM    | = *****     | 6-118                                                               |
| \$LAMDA   | = 000000    | #6-118                                                              |
| \$MVASC   | = ***** GX  | 7-186      11-293                                                   |
| \$NUMBR   | = 000002    | #6-118                                                              |
| \$OBUF    | = ***** GX  | *7-187      7-197      10-259      *10-261      11-282      *11-294 |
| \$OFDB    | = ***** GX  | *7-170      *7-199      *7-203                                      |
| \$RAD50   | = 000016    | #6-118                                                              |
| \$RONLY   | = *****     | 6-118      6-118                                                    |
| \$STRNG   | = 000004    | #6-118                                                              |
| \$SUBXP   | = 000010    | #6-118                                                              |
| \$TMLST   | = ***** GX  | 7-172                                                               |
| \$UPEVT   | = 000202 RG | #7-169                                                              |
| \$WDATA   | = ***** GX  | 7-200                                                               |
| \$\$\$FLG | = 177777    | #6-118                                                              |
| \$\$\$KEY | = 177777    | #6-118                                                              |
| .TPARS    | = ***** GX  | 7-193                                                               |

.TITLE UPDFEA - CFE UPDATE FEATURES DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE FEATURES DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

```

90 ;****
91 ; INITIALIZE TABLE GENERATION
92 ;****
93
94 000054 ISTAT$ LLCSTB,LLCKTB
95
96 000054 STATES$
97 000054 TRANS$ $RAD50,,SETNAM
98
99 000054 STATES$
100 000054 TRANS$ <','>,$MOVPT
101
102 000054 STATES$
103 000054 TRANS$!ZF
104
105 000054 STATES$
106 000054 TRANS$ "LLC"
107
108 000054 STATES$ BITS
109 000054 TRANS$!,BIT0
110 000054 TRANS$ $LAMDA,PRIOR,SETFLG
111
112 000054 STATES$ BIT0
113 000054 TRANS$!ZF
114
115 000054 STATES$
116 000054 TRANS$ "TIM",BITS
117 000054 TRANS$ "MFL",BITS
118 000054 TRANS$ "MTM",BITS
119 000054 TRANS$ "COU",BITS
120 000054 TRANS$ "X3P",BITS
121 000054 TRANS$ "SLI",BITS
122 000054 TRANS$ "INI",BITS
123
124 000054 STATES$ PRIOR
125 000054 TRANS$ <','>,$MOVPT
126
127 000054 STATES$
128 000054 TRANS$ $NUMBR,,SETPRI
129
130 000054 STATES$
131 000054 TRANS$ <','>,$MOVPT
132
133 000054 STATES$
134 000054 TRANS$ $NUMBR,,SETLIN
135
136 000054 STATES$
137 000054 TRANS$ $EOS,EXIT
138 000054 TRANS$ <','>,$MOVPT
139
140 000054 STATES$
141 000054 TRANS$ $STRNG,NLT,$MOVPT
142 000054 TRANS$ $LAMDA
143
144 000054 STATES$ NLT
145 000054 TRANS$ $EOS,EXIT
146 000054 TRANS$ <','>,$MOVPT

```

MACRO CROSS REFERENCE

CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$ | #5-94  | #5-110 | #5-117 | #5-118 | #5-119 | #5-120 | #5-121 | #5-122 | #5-138 | #5-142 |
|         | #5-146 | #5-150 | #5-154 | #5-158 | #5-162 |        |        |        |        |        |
| ISTAT\$ | #4-66  | 5-94   |        |        |        |        |        |        |        |        |
| LLCTP\$ | #4-66  | 4-68   |        |        |        |        |        |        |        |        |
| MSGDF\$ | #4-66  | 4-69   |        |        |        |        |        |        |        |        |
| MTRAN\$ | #5-94  |        |        |        |        |        |        |        |        |        |
| PRINT   | #4-57  |        |        |        |        |        |        |        |        |        |
| STATE\$ | #4-66  | 5-96   | #5-99  | #5-102 | #5-105 | #5-108 | #5-112 | #5-115 | #5-124 | #5-127 |
|         | #5-130 | #5-133 | #5-136 | #5-140 | #5-144 | #5-148 | #5-152 | #5-156 | #5-160 | #5-164 |
|         | #5-170 | #5-173 | #5-178 | #5-181 |        |        |        |        |        |        |
| TRANS   | #4-66  | #5-97  | #5-100 | #5-103 | #5-106 | #5-109 | #5-110 | #5-113 | #5-116 | #5-117 |
|         | #5-118 | #5-119 | #5-120 | #5-121 | #5-122 | #5-125 | #5-128 | #5-131 | #5-134 | #5-137 |
|         | #5-138 | #5-141 | #5-142 | #5-145 | #5-146 | #5-149 | #5-150 | #5-153 | #5-154 | #5-157 |
|         | #5-158 | #5-161 | #5-162 | #5-165 | #5-171 | #5-174 | #5-179 |        |        |        |



UPDNOD - CFE UPDATE NODE DEFINI MACRO V05.03b Saturday 29-Jun-85 05:53 <sup>C 13</sup>  
Table of contents

|     |     |                                  |
|-----|-----|----------------------------------|
| 6-  | 159 | \$UPNOD - UPDATE NODE DEFINITION |
| 7-  | 202 | SETNAM - SET UP NODE NAME        |
| 8-  | 227 | SETNID - SET UP NODE ID          |
| 9-  | 249 | SETNNA - SET EXEC AREA           |
| 9-  | 275 | SETNNM - SET NODE NUMBER         |
| 10- | 296 | SETHOA - SET HOST NODE AREA      |
| 10- | 322 | SETHOS - SET HOST NODE ADDRESS   |

\*\*FILE\*\*ID\*\*UPDOBJ

```

UU UU PPPPPPP DDDDDDDD 000000 88888888 JJ
UU UU PPPPPPP DDDDDDDD 000000 88888888 JJ
UU UU PP PP DD DD 00 00 88 88 JJ
UU UU PP PP DD DD 00 00 88 88 JJ
UU UU PP PP DD DD 00 00 88 88 JJ
UU UU PP PP DD DD 00 00 88 88 JJ
UU UU PPPPPPP DD DD 00 00 88888888 JJ
UU UU PPPPPPP DD DD 00 00 88888888 JJ
UU UU PP DD DD 00 00 88 88 JJ
UU UU PP DD DD 00 00 88 88 JJ
UU UU PP DD DD 00 00 88 88 JJ
UU UU PF DD DD 00 00 88 88 JJ
UUUUUUUUUU PF DDDDDDDD 000000 88888888 JJJJJJ
UUUUUUUUUU PP DDDDDDDD 000000 88888888 JJJJJJ

```

```

....
....
....
....

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

UPDOBJ      CREATED BY    MACRO    ON 29-JUN-85 AT 05:54      PAGE 2      C 15  
SYMBOL CROSS REFERENCE      CREF    04.00  
SYMBOL    VALUE            REFERENCES  
.TPARS    =    \*\*\*\*\*    GX      6-177

UPDPAR - CFE UPDATE PARTITION D MACRO V05.03b Saturday 29-Jun-85 05:54 Page 11  
 SETSBP - SETUP SBPOOL FLAG

```

299 .SBTTL SETSBP - SETUP SBPOOL FLAG
300 ;+
301 *** - SETSBP - SETUP SBPOOL FLAG
302 ;
303 INPUT:
304 (TEMP) = TEMPLATE ADDRESS
305 (TEMP)+C.STS = FLAGS WORD
306 ;
307 OUTPUT:
308 ;
309 ;
310 SETSBP:
311 MOV TEMP,R0 ; POINT AT TEMPLATE
312 MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
313 MOV #SBPSW,R1 ; POINT R1 AT INPUT STRING
314 MOV #SBPL,R2 ; AND LENGTH
315 CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
316 MOV RO,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
317 10$: RETURN
318 .END
319
311 000360
312 000360 016700 177406
313 000372 016700 000000G
314 000376 012701 000005'
315 000402 012702 000006
316 000406 004767 000000G
317 00412 010067 000000G
318 00416 000207
319
320 000001

```

```

173 000102 TRANS $NUMBR,EXIT,SETNCT
174
175 000102 STATES $ZF
176 000102 TRANS $ZF''
177
178 000102 STATES
179 000102 TRANS $.,$EXIT
180
181 ; EXIT STATE
182 ;
183 000102 STATES EXIT
184 000102 TRANS $EOS,$EXIT,$CLNUP
185
186 000102 STATES

```

.TITLE UPDDEC - CFE UPDATE DEC\$DF DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO UPDATE THE DEC\$DF MACRO

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

1.00 26-JUN-84  
MODULE CREATION  
5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

```

348
349
350
351
352
353
354
355
356
357
358
359
360 000410
361 000410 016702 177364
362 000414 005001
363 000416 156201 000021
364 000422 016700 000000G
365 000426 004767 000000G
366 000432 010067 000000G
367 000436 000207

 .SBTTL SETLST - LINK SERVICE THRESHOLD
 :
 : *** - SETLST - SETUP LINK SERVICE THRESHOLD
 :
 : INPUT:
 : TEMP = TEMPLATE ADDRESS
 : $OBUF = CURRENT POSITION IN OUTPUT BUFFER
 :
 : OUTPUT:
 :
 : -
 :
 SETLST:
 MOV TEMP,R2 ; GET STACK OF TEMPLATE
 CLR R1 ; GET LINK SERVICE THRESHOLD
 BISB C.LST(R2),R1 ;
 MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
 CALL $CNVD ; CONVERT NUMBER TO DECIMAL
 MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
 RETURN

```

```

161 .SBTTL $UPDLC - UPDATE DLC DEFINITION
162 +
163 *** - $UPDLC - UPDATE DLC DEFINITION
164 INPUT:
165 R3-R5 - TPARS REGISTERS
166 OUTPUT:
167 C-BIT = SUCCESS/FAILURE
168 -
169
170 $UPDLC::
171 MOV $TMLST,R0 ; POINT AT START OF TEMPLATES
172 10$: MOV R0,R1 ; SAVE PREVIOUS TEMPLATE ADDRESS
173 MOV (R0),R0 ; GET NEXT TEMPLATE
174 BEQ 10$,R0 ; IF EQ, END OF TEMPLATES
175 CMPB #CS.DLC,C.STS(R0) ; IS THIS AN LLC TEMPLATE ?
176 BNE 10$; IF NE, NO .. KEEP LOOKING
177
178 ; ASSUME THIS IS THE RIGHT TEMPLATE
179 MOV (R0),(R1) ; REMOVE TEMPLATE FROM CHAIN
180 MOV R0,TEMP ; SAVE TEMPLATE ADDRESS
181 BIT #CS.MOD,C.STS(R0) ; WAS THIS TEMPLATE MODIFIED ?
182 BNE 15$; IF NE, YES .. UPDATE THE RECORD
183 MOV #BUF1,$OFDB+F.NRBD+2 ; ELSE, OUTPUT THE INPUT BUFFER
184 BR 30$; AND LEAVE
185 15$: CALL $MOVST ; MOVE FIRST FIELD INTO OUTPUT BUFFER
186 MOV R5,-(SP) ; SAVE R5
187 CLR R1 ; IGNORE BLANKS
188 MOV #DLCKTB,R2 ; KEYWORD TABLE
189 MOV #DLCSTB,R5 ; STATE TABLE
190 CALL TPARS ; PARSE THE REST OF THE LINE
191 BCC 20$; NORMAL RETURN IF NO ERROR
192 INC $ERROR ; INDICATE SYNTAX ERROR
193 MOV (SP)+,R5 ; RESTORE R5
194 20$: RETURN
195 30$: RETURN
196
197 ; ERROR CONDITIONS
198 101$: JMP .CONER ; CONSISTENCY ERROR
199
200
201
202 000126 000167 000000G

```



```

57 .SBTTL MACRO CALLS AND LOCAL DATA
58 .SBTTL MACRO DEFINITIONS
59
60 ; LOCAL MACROS
61 ;
62
63 .MACRO PRINT TEXT
64 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
65 CALL $PRINT ; PRINT MESSAGE
66 .ENDM PRINT
67
68 .MACRO ERROR$ TEXT
69 ;
70 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
71 ; MESSAGE STRING.
72 ;
73 .IF DIF <TEXT><R0>
74 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
75 .ENDC
76 CALL $CFERR ; PRINT ERROR MESSAGE
77 .ENDM ERROR$
78
79 .MACRO EPRINT TEXT
80 ;
81 ; PRINT TEXT ON ERROR LUN
82 ;
83 .IF DIF <TEXT><R0>
84 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING .0 PRINT
85 .ENDC
86 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
87 .ENDM EPRINT
88
89 ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
90 ;
91 .MACRO $$$PTN R0U
92 .MACRO 'ROU OFS,OCTL,OPT,?L
93 .IF NB <OPT>
94 MOV OPT,R2
95 CALL $PTOPT
96 BCS L
97 .IFTF
98 MOV OFS,R2
99 .IF NB <OCTL>
100 MOV SP,R1
101 .IFF
102 CLR R1
103 .ENDC ; IF NB <OCTL>
104 CALL 'ROU
105 .IFT ; IF NB <OPT>
106
107 L:
108 .ENDC ; IF NB <OPT>
109 .ENDM
110 .ENDM $$$PTN
111 $$$PTN $PTNUM
112 $$$PTN $PTNMB
113
114 ;
115 ;
116 ;
117 ;
118 ;
119 ;
120 ;
121 ;
122 ;
123 ;
124 ;
125 ;
126 ;
127 ;
128 ;
129 ;
130 ;
131 ;
132 ;
133 ;
134 ;
135 ;
136 ;
137 ;
138 ;
139 ;
140 ;
141 ;
142 ;
143 ;
144 ;
145 ;
146 ;
147 ;
148 ;
149 ;
150 ;
151 ;
152 ;
153 ;
154 ;
155 ;
156 ;
157 ;
158 ;
159 ;
160 ;
161 ;
162 ;
163 ;
164 ;
165 ;
166 ;
167 ;
168 ;
169 ;
170 ;
171 ;
172 ;
173 ;
174 ;
175 ;
176 ;
177 ;
178 ;
179 ;
180 ;
181 ;
182 ;
183 ;
184 ;
185 ;
186 ;
187 ;
188 ;
189 ;
190 ;
191 ;
192 ;
193 ;
194 ;
195 ;
196 ;
197 ;
198 ;
199 ;
200 ;
201 ;
202 ;
203 ;
204 ;
205 ;
206 ;
207 ;
208 ;
209 ;
210 ;
211 ;
212 ;
213 ;
214 ;
215 ;
216 ;
217 ;
218 ;
219 ;
220 ;
221 ;
222 ;
223 ;
224 ;
225 ;
226 ;
227 ;
228 ;
229 ;
230 ;
231 ;
232 ;
233 ;
234 ;
235 ;
236 ;
237 ;
238 ;
239 ;
240 ;
241 ;
242 ;
243 ;
244 ;
245 ;
246 ;
247 ;
248 ;
249 ;
250 ;
251 ;
252 ;
253 ;
254 ;
255 ;
256 ;
257 ;
258 ;
259 ;
260 ;
261 ;
262 ;
263 ;
264 ;
265 ;
266 ;
267 ;
268 ;
269 ;
270 ;
271 ;
272 ;
273 ;
274 ;
275 ;
276 ;
277 ;
278 ;
279 ;
280 ;
281 ;
282 ;
283 ;
284 ;
285 ;
286 ;
287 ;
288 ;
289 ;
290 ;
291 ;
292 ;
293 ;
294 ;
295 ;
296 ;
297 ;
298 ;
299 ;
300 ;
301 ;
302 ;
303 ;
304 ;
305 ;
306 ;
307 ;
308 ;
309 ;
310 ;
311 ;
312 ;
313 ;
314 ;
315 ;
316 ;
317 ;
318 ;
319 ;
320 ;
321 ;
322 ;
323 ;
324 ;
325 ;
326 ;
327 ;
328 ;
329 ;
330 ;
331 ;
332 ;
333 ;
334 ;
335 ;
336 ;
337 ;
338 ;
339 ;
340 ;
341 ;
342 ;
343 ;
344 ;
345 ;
346 ;
347 ;
348 ;
349 ;
350 ;
351 ;
352 ;
353 ;
354 ;
355 ;
356 ;
357 ;
358 ;
359 ;
360 ;
361 ;
362 ;
363 ;
364 ;
365 ;
366 ;
367 ;
368 ;
369 ;
370 ;
371 ;
372 ;
373 ;
374 ;
375 ;
376 ;
377 ;
378 ;
379 ;
380 ;
381 ;
382 ;
383 ;
384 ;
385 ;
386 ;
387 ;
388 ;
389 ;
390 ;
391 ;
392 ;
393 ;
394 ;
395 ;
396 ;
397 ;
398 ;
399 ;
400 ;
401 ;
402 ;
403 ;
404 ;
405 ;
406 ;
407 ;
408 ;
409 ;
410 ;
411 ;
412 ;
413 ;
414 ;
415 ;
416 ;
417 ;
418 ;
419 ;
420 ;
421 ;
422 ;
423 ;
424 ;
425 ;
426 ;
427 ;
428 ;
429 ;
430 ;
431 ;
432 ;
433 ;
434 ;
435 ;
436 ;
437 ;
438 ;
439 ;
440 ;
441 ;
442 ;
443 ;
444 ;
445 ;
446 ;
447 ;
448 ;
449 ;
450 ;
451 ;
452 ;
453 ;
454 ;
455 ;
456 ;
457 ;
458 ;
459 ;
460 ;
461 ;
462 ;
463 ;
464 ;
465 ;
466 ;
467 ;
468 ;
469 ;
470 ;
471 ;
472 ;
473 ;
474 ;
475 ;
476 ;
477 ;
478 ;
479 ;
480 ;
481 ;
482 ;
483 ;
484 ;
485 ;
486 ;
487 ;
488 ;
489 ;
490 ;
491 ;
492 ;
493 ;
494 ;
495 ;
496 ;
497 ;
498 ;
499 ;
500 ;
501 ;
502 ;
503 ;
504 ;
505 ;
506 ;
507 ;
508 ;
509 ;
510 ;
511 ;
512 ;
513 ;
514 ;
515 ;
516 ;
517 ;
518 ;
519 ;
520 ;
521 ;
522 ;
523 ;
524 ;
525 ;
526 ;
527 ;
528 ;
529 ;
530 ;
531 ;
532 ;
533 ;
534 ;
535 ;
536 ;
537 ;
538 ;
539 ;
540 ;
541 ;
542 ;
543 ;
544 ;
545 ;
546 ;
547 ;
548 ;
549 ;
550 ;
551 ;
552 ;
553 ;
554 ;
555 ;
556 ;
557 ;
558 ;
559 ;
560 ;
561 ;
562 ;
563 ;
564 ;
565 ;
566 ;
567 ;
568 ;
569 ;
570 ;
571 ;
572 ;
573 ;
574 ;
575 ;
576 ;
577 ;
578 ;
579 ;
580 ;
581 ;
582 ;
583 ;
584 ;
585 ;
586 ;
587 ;
588 ;
589 ;
590 ;
591 ;
592 ;
593 ;
594 ;
595 ;
596 ;
597 ;
598 ;
599 ;
600 ;
601 ;
602 ;
603 ;
604 ;
605 ;
606 ;
607 ;
608 ;
609 ;
610 ;
611 ;
612 ;
613 ;
614 ;
615 ;
616 ;
617 ;
618 ;
619 ;
620 ;
621 ;
622 ;
623 ;
624 ;
625 ;
626 ;
627 ;
628 ;
629 ;
630 ;
631 ;
632 ;
633 ;
634 ;
635 ;
636 ;
637 ;
638 ;
639 ;
640 ;
641 ;
642 ;
643 ;
644 ;
645 ;
646 ;
647 ;
648 ;
649 ;
650 ;
651 ;
652 ;
653 ;
654 ;
655 ;
656 ;
657 ;
658 ;
659 ;
660 ;
661 ;
662 ;
663 ;
664 ;
665 ;
666 ;
667 ;
668 ;
669 ;
670 ;
671 ;
672 ;
673 ;
674 ;
675 ;
676 ;
677 ;
678 ;
679 ;
680 ;
681 ;
682 ;
683 ;
684 ;
685 ;
686 ;
687 ;
688 ;
689 ;
690 ;
691 ;
692 ;
693 ;
694 ;
695 ;
696 ;
697 ;
698 ;
699 ;
700 ;
701 ;
702 ;
703 ;
704 ;
705 ;
706 ;
707 ;
708 ;
709 ;
710 ;
711 ;
712 ;
713 ;
714 ;
715 ;
716 ;
717 ;
718 ;
719 ;
720 ;
721 ;
722 ;
723 ;
724 ;
725 ;
726 ;
727 ;
728 ;
729 ;
730 ;
731 ;
732 ;
733 ;
734 ;
735 ;
736 ;
737 ;
738 ;
739 ;
740 ;
741 ;
742 ;
743 ;
744 ;
745 ;
746 ;
747 ;
748 ;
749 ;
750 ;
751 ;
752 ;
753 ;
754 ;
755 ;
756 ;
757 ;
758 ;
759 ;
760 ;
761 ;
762 ;
763 ;
764 ;
765 ;
766 ;
767 ;
768 ;
769 ;
770 ;
771 ;
772 ;
773 ;
774 ;
775 ;
776 ;
777 ;
778 ;
779 ;
780 ;
781 ;
782 ;
783 ;
784 ;
785 ;
786 ;
787 ;
788 ;
789 ;
790 ;
791 ;
792 ;
793 ;
794 ;
795 ;
796 ;
797 ;
798 ;
799 ;
800 ;
801 ;
802 ;
803 ;
804 ;
805 ;
806 ;
807 ;
808 ;
809 ;
810 ;
811 ;
812 ;
813 ;
814 ;
815 ;
816 ;
817 ;
818 ;
819 ;
820 ;
821 ;
822 ;
823 ;
824 ;
825 ;
826 ;
827 ;
828 ;
829 ;
830 ;
831 ;
832 ;
833 ;
834 ;
835 ;
836 ;
837 ;
838 ;
839 ;
840 ;
841 ;
842 ;
843 ;
844 ;
845 ;
846 ;
847 ;
848 ;
849 ;
850 ;
851 ;
852 ;
853 ;
854 ;
855 ;
856 ;
857 ;
858 ;
859 ;
860 ;
861 ;
862 ;
863 ;
864 ;
865 ;
866 ;
867 ;
868 ;
869 ;
870 ;
871 ;
872 ;
873 ;
874 ;
875 ;
876 ;
877 ;
878 ;
879 ;
880 ;
881 ;
882 ;
883 ;
884 ;
885 ;
886 ;
887 ;
888 ;
889 ;
890 ;
891 ;
892 ;
893 ;
894 ;
895 ;
896 ;
897 ;
898 ;
899 ;
900 ;
901 ;
902 ;
903 ;
904 ;
905 ;
906 ;
907 ;
908 ;
909 ;
910 ;
911 ;
912 ;
913 ;
914 ;
915 ;
916 ;
917 ;
918 ;
919 ;
920 ;
921 ;
922 ;
923 ;
924 ;
925 ;
926 ;
927 ;
928 ;
929 ;
930 ;
931 ;
932 ;
933 ;
934 ;
935 ;
936 ;
937 ;
938 ;
939 ;
940 ;
941 ;
942 ;
943 ;
944 ;
945 ;
946 ;
947 ;
948 ;
949 ;
950 ;
951 ;
952 ;
953 ;
954 ;
955 ;
956 ;
957 ;
958 ;
959 ;
960 ;
961 ;
962 ;
963 ;
964 ;
965 ;
966 ;
967 ;
968 ;
969 ;
970 ;
971 ;
972 ;
973 ;
974 ;
975 ;
976 ;
977 ;
978 ;
979 ;
980 ;
981 ;
982 ;
983 ;
984 ;
985 ;
986 ;
987 ;
988 ;
989 ;
990 ;
991 ;
992 ;
993 ;
994 ;
995 ;
996 ;
997 ;
998 ;
999 ;
1000 ;

```

```

151 .SBTTL $UPDSC - UPDATE DESTINATION CALL PARAMETERS
152 +
153 *** - $UPDSC - UPDATE DESTINATION CALL PARAMETERS
154 :
155 INPUT:
156 R5 - TEMPLATE ADDRESS
157 :
158 OUTPUT:
159 C-BIT = SUCCESS/FAILURE
160 :
161 -
162
163 000010 $UPDSC::
164 000010 012767 000000G 000000C MOV #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
165 000016 010567 000000G MOV R5,.TEMP ; SAVE TEMPLATE ADDRESS
166 000022 005725 TST (R5)+ ; POINT TO C.STS
167 000024 010567 000000G MOV R5,.FLAGS ; SAVE ADDRESS OF FLAGS WORD
168 000030 012767 177777 000000G MOV #-1,.COMMA ; INITIALIZE FOR OPTIONS
169 000036 012700 000000G MOV #.BUFF,R0 ; POINT AT START OF OUTPUT BUFFER
170 000042 012701 000000' MOV #DSCDF,R1 ; POINT AT START OF DSC DEFINITION
171 000046 012702 000010 MOV #DSCLN,R2 ; GET LENGTH OF DSC DEFINITION
172 000052 004767 000000G CALL $MVASC ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
173 000056 010067 000000G MOV R0,$OBUF ; SAVE CURRENT POSITION IN OUTPUT BUFFER
174 000062 $PTR50 #C.MSK,#11.,#CS.MSK ; SET CALL MASK
175 000110 $PTR50 #C.VAL,#11.,#CS.VAL ; SET CALL VALUE
176 000136 016701 000000G MOV $OBUF,R1 ; CALCULATE LENGTH OF BUFFER
177 000142 162701 000000G SUB #.BUFF,R1 ;
178 000146 010167 000000C MOV R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
179 000152 004767 000000G CALL $WDATA ; WRITE THE RECORD TO THE OUTPUT FILE
180 000156 012767 000000G 000000C MOV # $BUFI,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE 'END$DF' LINE
181 000164 000207 RETURN
182
183 000001 END
184

```

UPDDST      CREATED BY    MACRO    ON 29-JUN-85 AT 05:51      PAGE 2      D 7

MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES                              |
|------------|-----------------------------------------|
| EPRINT     | #4-79                                   |
| ERROR\$    | #4-68                                   |
| PRINT      | #4-63                                   |
| \$PTNMB    | #4-112      6-179      6-180            |
| \$PTNUM    | #4-111                                  |
| \$PTR50    | #4-116      6-177      6-178      6-181 |
| \$\$\$PTN  | #4-91      4-111      4-112             |

|     |     |                                  |
|-----|-----|----------------------------------|
| 4-  | 56  | MACRO DEFINITIONS                |
| 5-  | 75  | LOCAL DATA                       |
| 6-  | 111 | STATE TABLES                     |
| 7-  | 157 | \$UPEVT - SETUP EVENT DEFINITION |
| 8-  | 206 | SETCLS - WRITE OUT EVENT CLASS   |
| 9-  | 225 | SETMSK - WRITE OUT EVENT MASK    |
| 10- | 246 | SETNUM - WRITE OUT NUMBER        |
| 11- | 264 | SETSNK - WRITE OUT SINK NAME     |

UPDEV T      CREATED BY    MACRO    ON 29-JUN-85 AT 05:51      PAGE 3      D 9

MACRO CROSS REFERENCE

CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$ | #6-118 | #6-153 |        |        |        |        |        |        |        |        |
| FLTDF\$ | #4-71  | 4-73   |        |        |        |        |        |        |        |        |
| ISTAT\$ | #4-71  | 6-118  |        |        |        |        |        |        |        |        |
| MTRAN\$ | #6-118 |        |        |        |        |        |        |        |        |        |
| PRINT   | #4-62  |        |        |        |        |        |        |        |        |        |
| STATE\$ | #4-71  | 6-121  | #6-124 | #6-127 | #6-130 | #6-133 | #6-136 | #6-142 | #6-148 | #6-151 |
|         | #6-155 |        |        |        |        |        |        |        |        |        |
| TRAN\$  | #4-71  | #6-122 | #6-125 | #6-128 | #6-131 | #6-134 | #6-137 | #6-143 | #6-149 | #6-152 |
|         | #6-153 |        |        |        |        |        |        |        |        |        |

```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 ;****
65 ; MACRO CALLS
66 ;****
67
68 .MCALL ISTAT$,STAT$,TRANS
69
70 ;****
71 ; LOCAL DATA
72 ;****
73
74 000000 TEMP: .BLKW 1 ; ADDRESS OF FEATURES TEMPLATE
75 ;
76 ; DEFINE FEATURES TEMPLATE
77 ;
78 000000 .ASECT
79 000000 .=0
80 000000 .BLKW 1 ; C.LNK
81 000002 .BLKW 1 ; C.STS
82 000004 C.DEV: .BLKW 1 ; 2 CHARACTER DEVICE NAME
83 000006 C.WD1: .BLKW 1 ; INITIAL FEATURES
84 000010 C.WD2: .BLKW 1 ; STATES SETTABLE
85 000012 C.WD3: .BLKW 1 ; CURRENT STATES
86 000014 C.LEN=.
87 000002 .PSECT
88

```

```

147
148 000054 STATES
149 000054 TRANS $NUMBR,EXT,SETNLT
150 000054 TRANS $LAMDA
151
152 000054 STATES EXT
153 000054 TRANS $EOS,EXIT
154 000054 TRANS <','>,$MOVPT
155
156 000054 STATES
157 000054 TRANS $NUMBR,CTIM,SETEXT
158 000054 TRANS $LAMDA
159
160 000054 STATES CTIM
161 000054 TRANS $EOS,EXIT
162 000054 TRANS <','>,$MOVPT
163
164 000054 STATES
165 000054 TRANS $NUMBR,EXIT,SETCTM
166
167 ;
168 ; SUB-EXPRESSION TO PARSE "ZF."
169 ;
170 000054 STATES ZF
171 000054 TRANS "ZF"
172
173 000054 STATES
174 000054 TRANS '.,$EXIT
175
176 ; EXIT STATE
177 ;
178 000054 STATES EXIT
179 000054 TRANS $EOS,$EXIT,$CLNUP
180
181 000054 STATES

```

\*\*FILE\*\*ID\*\*UPDLOG

```

UU UU PPPPPPP DDDDDDD LL 000000 GGGGGGG
UU UU PPPPPPP DDDDDDD LL 000000 GGGGGGG
UU UU PP DD LL 00 00 GG
UU UU PP DD LL 00 00 GG
UU UU PP DD LL 00 00 GG
UU UU PP DD LL 00 00 GG
UU UU PP DD LL 00 00 GG
UU UU PPPPPPP DD DD LL 00 00 GG
UU UU PPPPPPP DD DD LL 00 00 GG
UU UU PP DD DD LL 00 00 GG
UU UU PP DD DD LL 00 00 GG
UU UU PP DD DD LL 00 00 GG
UU UU PP DD DD LL 00 00 GG
UU UU PP DD DD LL 00 00 GG
UU UU PP DD DD LL 00 00 GG
UUUUUUUU PP DDDDDDD LLLLLLLLL 000000 GGGGGG
UUUUUUUU PP DDDDDDD LLLLLLLLL 000000 GGGGGG

```

....  
....  
....  
....

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LLLLLLLL SSSSSSS TT
LLLLLLLL SSSSSSS TT

```



.TITLE UPDNOD - CFE UPDATE NODE DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE NODE DEFINITIONS

#### DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

UPDOBJ - CFE UPDATE OBJECT DEFIN MACRO V05.03b Saturday 29-Jun-85 05:54 <sup>D 14</sup>  
Table of contents

|     |     |                                                         |
|-----|-----|---------------------------------------------------------|
| 6-  | 141 | \$UPOBJ - SETUP OBJECT DEFINITION                       |
| 7-  | 190 | SETYPE - WRITE OUT OBJECT TYPE                          |
| 8-  | 214 | SETNAM - WRITE OUT OBJECT NAME                          |
| 9-  | 247 | SETVFY - SETUP OBJECT VERIFICATION LEVEL                |
| 10- | 272 | SETFLG - WRITE OUT THE FLAGS BYTE                       |
| 11- | 297 | SETCPY - WRITE OUT THE MAXIMUM NUMBER OF COPIES ALLOWED |

UPDOBJ      CREATED BY    MACRO    ON 29-JUN-85 AT 05:54      PAGE 3      D 15  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-104     |        |        |        |        |        |        |        |        |        |
| ISTAT\$    | #4-68      | 5-104  |        |        |        |        |        |        |        |        |
| MTRAN\$    | #5-104     |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-59      |        |        |        |        |        |        |        |        |        |
| STAT\$     | #4-68      | 5-107  | #5-110 | #5-113 | #5-116 | #5-119 | #5-122 | #5-125 | #5-128 | #5-131 |
|            | #5-136     | #5-139 |        |        |        |        |        |        |        |        |
| TRAN\$     | #4-63      | #5-108 | #5-111 | #5-114 | #5-117 | #5-120 | #5-123 | #5-126 | #5-129 | #5-132 |
|            | #5-137     |        |        |        |        |        |        |        |        |        |

UPDPAR - CFE UPDATE PARTITION D MACRO V05.03b Saturday 29-Jun-85 05:54 Page 11-1

Symbol table

|         |          |     |                    |          |     |          |          |     |          |          |              |          |     |
|---------|----------|-----|--------------------|----------|-----|----------|----------|-----|----------|----------|--------------|----------|-----|
| ALLC    | 000104R  | 002 | PARSTB             | 000000RG | 002 | TOPSW    | 000002R  | 002 | \$DNUMB= | 000014   | \$RAD50=     | 000016   |     |
| CS.PAR= | ***** GX |     | SBCHK              | 000120R  | 002 | UMRS     | 000056R  | 002 | \$EOS =  | 000012   | \$STRNG=     | 000004   |     |
| CS.TOP= | ***** GX |     | SBPL =             | 000006   |     | \$ALPHA= | 000022   |     | \$ERROR= | ***** GX | \$SUBXP=     | 000010   |     |
| C.ALL   | 000016   |     | SBPSW              | 000005R  |     | \$ANY =  | 000020   |     | \$EXIT = | 000000   | \$TMLST=     | ***** GX |     |
| C.EXT   | 000014   |     | SETALL             | 000240R  |     | \$BLANK= | 000006   |     | \$FAIL = | 177777   | \$UPPAR      | 000014RG |     |
| C.LEN = | 000020   |     | SETNMO             | 000106R  |     | \$BYTMN= | ***** GX |     | \$LAMDA= | 000000   | \$\$\$\$FLG= | 177777   |     |
| C.NAMO  | 000004   |     | SETNM1             | 000170R  |     | \$BYTXT= | ***** GX |     | \$MOVPT= | ***** GX | \$\$\$\$KEY= | 000002   |     |
| C.NAM1  | 000010   |     | SETSBP             | 000366R  |     | \$CLNUP= | ***** GX |     | \$MOVST= | ***** GX | \$\$\$\$STA= | 000000   |     |
| C.STS = | ***** GX |     | SETTOP             | 000324R  |     | \$CNVD = | ***** GX |     | \$MVASC= | ***** GX | \$\$\$\$TMP= | 000011R  | 004 |
| EXIT    | 000136R  | 002 | TEMP               | 000010R  | 002 | \$CSTA = | ***** GX |     | \$NUMBR= | 000002   | .CONER=      | ***** GX |     |
| NEXT    | 000012R  | 002 | TOP                | 000044R  | 002 | \$DIGIT= | 000024   |     | \$OBUF = | ***** GX | .TPARS=      | ***** GX |     |
| PARKTB  | 000000RG | 003 | TOPL =             | 000003   |     |          |          |     |          |          |              |          |     |
| . ABS.  | 000020   | 000 | (RW,I,GBL,ABS,OVR) |          |     |          |          |     |          |          |              |          |     |
|         | 000420   | 001 | (RW,I,LCL,REL,CON) |          |     |          |          |     |          |          |              |          |     |
| \$STATE | 000144   | 002 | (RW,D,LCL,REL,CON) |          |     |          |          |     |          |          |              |          |     |
| \$KTAB  | 000006   | 003 | (RW,D,LCL,REL,CON) |          |     |          |          |     |          |          |              |          |     |
| \$KSTR  | 000020   | 004 | (RW,D,LCL,REL,CON) |          |     |          |          |     |          |          |              |          |     |

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10349 Words ( 41 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:29.37

SY:UPDPAR.V2,[132,134]UPDPAR/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDPAR

```

188 .SBTTL $UPDDM - UPDATE DDM DEFINITION
189 +
190 *** - $UPDDM - UPDATE DDM DEFINITION
191 :
192 INPUT:
193 R3-R5 - TPARS REGISTERS
194 :
195 OUTPUT:
196 C-BIT = SUCCESS/FAILURE
197 :
198 -
199
200 000102 $UPDDM::
201 000102 012700 000000G 10$: MOV #$TMLST,R0 ; POINT AT START OF TEMPLATES
202 000106 010001 MOV R0,R1 ; SAVE ADDRESS OF CURRENT TEMPLATE
203 000110 011000 MOV (R0),R0 ; GET NEXT TEMPLATE
204 000112 001436 BEQ 101$; IF EQ, END OF TEMPLATES
205 000114 122760 000000G 000000G CMPB #CS.DDM,C.STS(R0) ; IS THIS A DDM TEMPLATE ?
206 000122 001371 BNE 10$; IF NE, NO .. KEEP LOOKING
207 :
208 ; ASSUME THIS IS THE RIGHT TEMPLATE
209 :
210 000124 011011 MOV (R0),(R1) ; REMOVE TEMPLATE FROM CHAIN
211 000126 010067 177646 MOV R0,TEMP ; SAVE TEMPLATE ADDRESS
212 000132 032760 000000G 000000G BIT #CS.MOD,C.STS(R0) ; WAS THIS TEMPLATE MODIFIED ?
213 000140 001004 BNE 15$; IF NE, YES .. UPDATE THE RECORD
214 000142 012767 000000G 000000G MOV #SBUF1,$OFDB+F.NRBD+2 ; ELSE, OUTPUT THE INPUT BUFFER
215 000150 000416 BR 30$; AND LEAVE
216 000152 004767 000000G 15$: CALL $MOVST ; MOVE FIRST FIELD INTO OUTPUT BUFFER
217 000156 010546 MOV R5,-(SP) ; SAVE R5
218 000160 005001 CLR R1 ; IGNORE BLANK
219 000162 012702 000000' MOV #DDMKT8,R2 ; KEYWORD TABLE
220 000166 012705 000000' MOV #DDMST8,R5 ; STATE TABLE
221 000172 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
222 000176 103002 BCC 20$; NORMAL RETURN IF NO ERROR
223 000200 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
224 000204 012605 MOV (SP)+,R5 ; RESTORE R5
225 000206 000207 20$: 30$: RETURN
226 :
227 ; ERROR CONDITIONS
228 :
229 000210 000167 000000G 101$: JMP .CONER ; CONSISTENCY ERROR

```

UF  
SE

```

369
370
371
372
373
374
375
376
377
378
379
380
381 000440
382 000440 016702 177334
383 000444 005001
384 000446 156201 000015
385 000452 016700 000000G
386 000456 004767 000000G
387 000462 010067 000000G
388 000466 000207

 .SBTTL SETRET - RETRANSMIT FACTOR
 ;+
 *** - SETRET - SETUP RETRANSMIT FACTOR
 :
 INPUT:
 : TEMP = TEMPLATE ADDRESS
 : $OBUF = CURRENT POSITION IN OUTPUT BUFFER
 :
 OUTPUT:
 :-
 SETRET:
 MOV TEMP,R2 ; GET START OF TEMPLATE
 CLR R1 ; GET RETRANSMIT FACTOR
 BISB C,RETF(R2),R1 ;
 MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
 CALL $CNVD ; CONVERT NUMBER TO DECIMAL
 MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
 RETIRN

```

```

204 .SBTTL SETNAM - CHECK PROCESS NAME
205 ;+
206 ;*** - SETNAM - CHECK PROCESS NAME
207 ;
208 ; INPUT:
209 ; .PSTPT = START ADDRESS OF THE PROCESS NAME IN ASCII
210 ; .PSTCN = NUMBER OF CHARACTERS IN THE PROCESS NAME
211 ;
212 ; OUTPUT:
213 ; C-BIT = SUCCESS/FAILURE
214 ;
215 ; -
216
217 000132 SETNAM: CALL $MOVPT ; MOVE THE PROCESS NAME INTO THE OUTPUT BUFFER
218 000132 MOV .PSTPT,R0 ; GET ADDRESS OF ASCII PROCESS NAME
219 000136 MOV PC,R1 ; PERIODS ARE ACCEPTABLE
220 000142 MOV $CAT5 ; CONVERT PROCESS NAME TO RAD50
221 000144 CALL $CAT5 ; GET START OF TEMPLATE
222 000150 MOV TEMP,R0 ; IS THIS THE RIGHT TEMPLATE ?
223 000154 CMP R1,C.NAM(R0) ; IF EQ, YES -- OKAY
224 000160 BEQ 10$; ELSE, REJECT THE TRANSITION
225 000162 ADD #2,(SP)
226 000166 000207
 10$: RETURN

```



```

114 ; PUT RAD50 TEMPLATE FIELD TO BUFFER
115 ;
116 .MACRO $PTR50 OFS,WC,OPT,?L
117 .IF NB <OPT>
118 MOV OPT,R2
119 CALL $PTOPT
120 BCS L
121 .IFTF
122 MOV OFS,R2
123 MOV WC,R1
124 CALL $PTR50
125 .IFT
126 L:
127 .ENDC
128 .ENDM
129

```

UPDDSC - UPDATE DESTINATION CAL MACRO V05.03b Saturday 29-Jun-85 05:50 Page 6-1  
Symbol table

|                  |                  |                   |                   |                  |
|------------------|------------------|-------------------|-------------------|------------------|
| CS.MSK= ***** GX | C.VAL 000032     | \$BUFI = ***** GX | \$PTOPT= ***** GX | .BUFF = ***** GX |
| CS.VAL= ***** GX | DSCDF 000000R    | \$MVASC= ***** GX | \$PTR50= ***** GX | .COMMA= ***** GX |
| C.LEN = 000060   | DSCLN = 000010   | \$OBUF = ***** GX | \$UPDSC 000010RG  | .FLAGS= ***** GX |
| C.MSK 000004     | F.NRBD= ***** GX | \$OFDB = ***** GX | \$WDATA= ***** GX | .TEMP = ***** GX |

. ABS. 000060 000 (RW,I,GBL,ABS,OVR)  
000166 001 (RW,I,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 8830 Words ( 35 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:05.20  
SY:UPDDSC.V2,[132,134]UPDDSC/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDDSC

|          |    |        |        |        |          |          |        |
|----------|----|--------|--------|--------|----------|----------|--------|
| UU       | UU | PPPPPP | DDDDDD | DDDDDD | TTTTTTTT | EEEEEEEE |        |
| UU       | UU | PPPPPP | DDDDDD | DDDDDD | TTTTTTTT | EEEEEEEE |        |
| UU       | UU | PP     | DD     | DD     | DD       | TT       | EE     |
| UU       | UU | PP     | DD     | DD     | DD       | TT       | EE     |
| UU       | UU | PP     | DD     | DD     | DD       | TT       | EE     |
| UU       | UU | PP     | DD     | DD     | DD       | TT       | EE     |
| UU       | UU | PPPPPP | DD     | DD     | DD       | TT       | EEEEEE |
| UU       | UU | PPPPPP | DD     | DD     | DD       | TT       | EEEEEE |
| UU       | UU | PP     | DD     | DD     | DD       | TT       | EE     |
| UU       | UU | PP     | DD     | DD     | DD       | TT       | EE     |
| UU       | UU | PP     | DD     | DD     | DD       | TT       | EE     |
| UU       | UU | PP     | DD     | DD     | DD       | TT       | EE     |
| UUUUUUUU | PP | DDDDDD | DDDDDD | DDDDDD | TT       | EEEEEEEE | ....   |
| UUUUUUUU | PP | DDDDDD | DDDDDD | DDDDDD | TT       | EEEEEEEE | ....   |

|          |        |          |
|----------|--------|----------|
| LL       | SSSSSS | TTTTTTTT |
| LL       | SSSSSS | TTTTTTTT |
| LL       | SS     | TT       |
| LL       | SS     | TT       |
| LL       | SS     | TT       |
| LL       | SS     | TT       |
| LL       | SSSSSS | TT       |
| LL       | SSSSSS | TT       |
| LL       | SS     | TT       |
| LL       | SS     | TT       |
| LL       | SS     | TT       |
| LL       | SS     | TT       |
| LLLLLLLL | SSSSSS | TT       |
| LLLLLLLL | SSSSSS | TT       |

.TITLE UPDEVT - CFE UPDATE EVENT DEFINITION IN CETAB  
 .IDENT /V05.00/  
 .NLIST BEX

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE EVENT DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 07-JUL-80  
 DECNET-11M/S V3.0  
 DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0

\*\*FILE\*\*ID\*\*UPDFIL

```

UU UU PPPPPPP DDDDDDDD FFFFFFFF IIIIII LL
UU UU PPPPPPP DDDDDDDD FFFFFFFF IIIIII LL
UU UU PP DD FF II LL
UU UU PP DD FF II LL
UU UU PP DD FF II LL
UU UU PP DD FF II LL
UU UU PP DD FF II LL
UU UU PPPPPPP DI DD FFFFFFFF II LL
UU UU PPPPPPP DD DD FFFFFFFF II LL
UU UU PP DD DD FF II LL
UU UU PP DD DD FF II LL
UU UU PP DD DD FF II LL
UU UU PP DD DD FF II LL
UUUUUUUU PP DDDDDDDD FF IIIIII LLLLLLLLLL
UUUUUUUU PP DDDDDDDD FF IIIIII LLLLLLLLLL

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

```

90
91 ;****
92 ; INITIALIZE TABLE GENERATION
93 ;****
94
95 000002 ISTAT$ FEASTB,FEAKTB
96
97
98 000002 STATES$
99 000002 TRANS$ $STRNG,, $MOVPT
100
101 000002 STATES$
102 000002 TRANS$ <'>,, $MOVPT
103
104 000002 STATES$
105 000002 TRANS$ $NUMBR,, $MOVPT
106
107 000002 STATES$
108 000002 TRANS$ <'>,, $MOVPT
109
110 000002 STATES$
111 000002 TRANS$ $NUMBR,, $MOVPT
112
113 000002 STATES$
114 000002 TRANS$ <'>,, $MOVPT
115
116 000002 STATES$
117 000002 TRANS$ $NUMBR,, SETWD3
118
119 ; EXIT STATE
120 ;
121 000002 STATES$ EXIT
122 000002 TRANS$ $EOS,$EXIT,$CLNUP
123
124 000002 STATES$

```

```

183 .SBTTL $UPLLC - UPDATE LLC DEFINITION.
184 *
185 *** - $UPLLC - UPDATE LLC DEFINITION
186
187 INPUT:
188 R3-R5 - TPARS REGISTERS
189
190 OUTPUT:
191 C-BIT = SUCCESS/FAILURE
192
193 -
194
195 $UPLLC::
196 000054 012700 000000G MOV #$TMLST,R0 ; POINT AT START OF TEMPLATES
197 000054 010001 10$: MOV R0,R1 ; SAVE PREVIOUS TEMPLATE ADDRESS
198 000062 011000 MOV (R0),R0 ; GET NEXT TEMPLATE
199 000064 001436 BEQ 101$; IF EQ, CANNOT FIND LLC TEMPLATE
200 000066 122760 000000G 000000G CMPB #CS.LLC,C.STS(R0) ; IS THIS AN LLC TEMPLATE ?
201 000074 001371 BNE 10$; IF NE, NO .. KEEP LOOKING
202
203 ; ASSUME THIS IS THE RIGHT TEMPLATE
204
205 000076 011011 MOV (R0),(R1) ; REMOVE TEMPLATE FROM CHAIN
206 000100 010067 177674 MOV R0,TEMP ; SAVE TEMPLATE ADDRESS
207 000104 032760 000000G 000000G BIT #CS.MOD,C.STS(R0) ; WAS THIS TEMPLATE MODIFIED ?
208 000112 001004 BNE 15$; IF NE, YES .. UPDATE THE RECORD
209 000114 012767 000000G 000000C MOV #SBUF1,$OFDB+F.NRBD+2 ; ELSE, OUTPUT THE INPUT BUFFER
210 000122 000416 BR 30$; AND LEAVE
211 000124 004767 000000G 15$: CALL $MOVST ; MOVE FIRST FIELD INTO OUTPUT BUFFER
212 000130 010546 MOV R5,-(SP) ; SAVE R5
213 000132 005001 CLR R1 ; IGNORE BLANKS
214 000134 012702 000000' MOV #LLCKTB,R2 ; KEYWORD TABLE
215 000140 012705 00J000' MOV #LLCSTB,R5 ; STATE TABLE
216 000144 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
217 000150 103002 BCC 20$; NORMAL RETURN IF NO ERROR
218 000152 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
219 000156 012605 20$: MOV (SP)+,R5 ; RESTORE R5
220 000160 000207 30$: RETURN
221
222 ; ERROR CONDITIONS
223
224 000162 000167 000000G 101$: JMP .CONER ; CONSISTENCY ERROR

```

UPDLOG - CFE UPDATE LOGGING STA MACRO V05.03b Saturday 29-Jun-85 05:53 <sup>E 12</sup>  
Table of contents

|    |     |                                         |
|----|-----|-----------------------------------------|
| 4- | 55  | MACRO DEFINITIONS                       |
| 5- | 92  | LOCAL DATA                              |
| 6- | 111 | STATE TABLES                            |
| 7- | 129 | \$UPLOG - UPDATE LOGGING STATE TEMPLATE |
| 8- | 175 | SETNUM - SET THE NEXT NUMBER            |



```

56 ;****
57 ; LOCAL MACROS
58 ;****
59
60 .MACRO PRINT TEXT
61 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
62 CALL $PRINT ; PRINT MESSAGE
63 .ENDM PRINT
64
65 ;****
66 ; MACRO CALLS
67 ;****
68
69 .MCALL ISTAT$,STAT$,TRANS$
70
71 ;****
72 ; LOCAL DATA
73 ;****
74
75 000074 LA = '< ; LEFT ANGLE BRACKET
76 000076 RA = '> ; RIGHT ANGLE BRACKET
77 000000 TEMP: .BLKW 1 ; ADDRESS OF NODE TEMPLATE
78
79 ;
80 ; DEFINE NODE TEMPLATE
81 ;
82 .ASECT
83 .=0 000000
84 .BLKW 1 ; C.LNK
85 .BLKW 1 ; C.STS
86 C.NAM: .BLKW 1 ; NODE NAME (RAD50)
87 .BLKW 1
88 C.NID: .BLKW 1 ; NODE ID
89 .BLKB 32. ; LENGTH AND STRING
90 C.NUM: .BLKW 1 ; NODE ADDRESS
91 C.HOS: .BLKW 1 ; HOST NODE ADDRESS
92 C.LEN=. ;
93 000002 .PSECT
94

```

.TITLE UPDOBJ - CFE UPDATE OBJECT DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE OBJECT DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/Rsx V1.0

\*\*FILE\*\*ID\*\*UPDPAR

```

UU UU PPPPPPP DDDDDDD PPPPPPP AAAAAA RRRRRRR
UU UU PPPPPPP DDDDDDD PPPPPPP AAAAAA RRRRRRR
UU UU PP PP DD DD PP PP AA AA RR RR
UU UU PP PP DD DD PP PP AA AA RR RR
UU UU PP PP DD DD PP PP AA AA RR RR
UU UU PPPPPPP DD DD PPPPPPP AA AA RRRRRRR
UU UU PPPPPPP DD DD PPPPPPP AA AA RRRRRRR
UU UU PP DD DD PP AAAAAAAAAA RR RR
UU UU PP DD DD PP AAAAAAAAAA RR RR
UU UU PP DD DD PP AA AA RR RR
UU UU PP DD DD PP AA AA RR RR
UUUUUUUU PP DDDDDDD PP AA AA RR RR
UUUUUUUU PP DDDDDDD PP AA AA RR RR

```

```

....
....
....
....

```

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLL SSSSSSS TT
LLLLLLLL SSSSSSS TT

```

| SYMBOL    | VALUE      | REFERENCES |
|-----------|------------|------------|
| CS.PAR    | = ***** GX | 6-164      |
| CS.TOP    | = ***** GX | 10-289     |
| C.ALL     | 000016     | #4-81      |
| C.EXT     | 000014     | #4-80      |
| C.LEN     | = 000020   | #4-82      |
| C.NAM0    | 000004     | #4-78      |
| C.NAM1    | 000010     | #4-79      |
| C.STS     | = ***** GX | 6-164      |
| PARKTB    | 000000 RG  | #5-90      |
| PARSTB    | 000000 RG  | 6-174      |
| SBPL      | = 000006   | #5-90      |
| SBPSW     | 000005 R   | 6-175      |
| SETALL    | 000240 R   | 11-315     |
| SETNMO    | 000106 R   | 4-69       |
| SETNMI    | 000170 R   | 11-314     |
| SETSBP    | 000366 R   |            |
| SETTOP    | 000324 R   |            |
| TEMP      | 000000 R   |            |
| TOPL      | = 000003   | *6-170     |
| TOPSW     | 000002 R   | 7-200      |
| \$ALPHA   | = 000022   | 7-205      |
| \$ANY     | = 000020   | 7-208      |
| \$BLANK   | = 000006   | 8-228      |
| \$BYTMN   | = ***** GX | 8-233      |
| \$BYTXT   | = ***** GX | 10-288     |
| \$CNVD    | = ***** GX | 11-312     |
| \$CSTA    | = ***** GX |            |
| \$DIGIT   | = 000024   |            |
| \$DNUMB   | = 000014   |            |
| \$EOS     | = 000012   |            |
| \$ERROR   | = ***** GX |            |
| \$EXIT    | = 000000   |            |
| \$FAIL    | = 177777   |            |
| \$GPRM    | = *****    |            |
| \$LAMDA   | = 000000   |            |
| \$MOVST   | = ***** GX |            |
| \$MVASC   | = ***** GX |            |
| \$NUMBR   | = 000002   |            |
| \$OBUF    | = ***** GX |            |
| \$RAD50   | = 000016   |            |
| \$RONLY   | = *****    |            |
| \$STRNG   | = 000004   |            |
| \$SUBXP   | = 000010   |            |
| \$TMLST   | = ***** GX |            |
| \$UPPAR   | 000014 RG  |            |
| \$\$\$FLG | = 177777   |            |
| \$\$\$KEY | = 177777   |            |
| .CONER    | = ***** GX |            |
| .TPARS    | = ***** GX |            |

SETNAM - CHECK PROCESS NAME

```

231 .SBTTL SETNAM - CHECK PROCESS NAME
232 ;
233 ; * - SETNAM - CHECK PROCESS NAME
234 ;
235 ; INPUT:
236 ; .PSTPT = START ADDRESS OF THE PROCESS NAME IN ASCII
237 ; .PSTCN = NUMBER OF CHARACTERS IN THE PROCESS NAME
238 ;
239 ; OUTPUT:
240 ; C-BIT = SUCCESS/FAILURE
241 ;
242 ; -
243 ;
244 SETNAM:
245 000214 004767 000000G CALL $MOVPT ; MOVE THE PROCESS NAME INTO THE OUTPUT BUFFER
246 000220 016700 000000G MOV .PSTPT,R0 ; GET ADDRESS OF ASCII PROCESS NAME
247 000224 010701 000000G MOV PC,R1 ; PERIODS ARE ACCEPTABLE
248 000226 004767 000000G CALL $CAT5 ; CONVERT PROCESS NAME TO RAD50
249 000232 016700 177542 MOV TEMP,R0 ; GET START OF TEMPLATE
250 000236 020160 000004 CMP R1,C.NAM(R0) ; IS THIS THE RIGHT TEMPLATE ?
251 000242 001402 000000G BEQ 10$; IF EQ, YES .. OKAY
252 000244 062716 000002 ADD #2,(SP) ; ELSE, REJECT THE TRANSITION
253 000250 000207 000000G 10$: RETURN

```

SETFLG - SETUP THE FLAGS WORD

```

61 .SBTTL DEC$DF PARSE TABLES
62
63 ;
64 ; INITIALIZE TABLE GENERATION
65 ;
66
67 000002 ISTAT$ DECSTB,DECKTB
68
69 000002 STATES$; MAXIMUM LOGICAL LINKS
70 000002 TRANS$ $NUMBR,,SETLKS
71
72 000002 STATES$
73 000002 TRANS$ <','>,$MOVPT
74
75 000002 STATES$; MAXIMUM NODE COUNTERS
76 000002 TRANS$ $NUMBR,,SETNDC
77
78 000002 STATES$
79 000002 TRANS$ <','>,$MOVPT
80
81 000002 STATES$
82 000002 TRANS$ $NUMBR,,SETINC ; INCOMING TIMER
83
84 000002 STATES$
85 000002 TRANS$ <','>,$MOVPT
86
87 000002 STATES$
88 000002 TRANS$ $NUMBR,,SETOUT ; OUTGOING TIMER
89
90 000002 STATES$
91 000002 TRANS$ <','>,$MOVPT
92
93 000002 STATES$
94 000002 TRANS$ $NUMBR,,SETINA ; INACTIVITY TIMER
95
96 000002 STATES$
97 000002 TRANS$ <','>,$MOVPT
98
99 000002 STATES$
100 000002 TRANS$ $NUMBR,,SETDEF ; DELAY FACTOR
101
102 000002 STATES$
103 000002 TRANS$ <','>,$MOVPT
104
105 000002 STATES$
106 000002 TRANS$ $NUMBR,,SETDEW ; DELAY WEIGHT
107
108 000002 STATES$
109 000002 TRANS$ <','>,$MOVPT
110
111 000002 STATES$
112 000002 TRANS$ $NUMBR,,SETIPL ; INPUT PACKET LIMITER
113
114 000002 STATES$
115 000002 TRANS$ <','>,$MOVPT
116
117 000002 STATES$

```

```

390 .SBTTL SETSEG - ECL SEGMENT SIZE
391 ;+
392 *** - SETSEG - SETUP ECL SEGMENT SIZE
393 ;
394 INPUT:
395 TEMP = TEMPLATE ADDRESS
396 $OBUF = CURRENT POSITION IN OUTPUT BUFFER
397 ;
398 OUTPUT:
399 ;
400 ;
401 ;
402 000470 SETSEG:
403 000470 016702 177304 MOV TEMP,R2 ; GET START OF TEMPLATE
404 000474 016201 000004 MOV C.SEG(R2),R1 ; GET ECL SEGMENT SIZE
405 000500 001402 BEQ 5$; IF EQ BRANCH
406 000502 162701 000022 SUB #N$OVR,R1 ; ELSE REMOVE INTERNAL BIAS
407 000506 016700 000000G 5$: MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
408 000512 004767 000000G CALL $CNVD ; CONVERT NUMBER TO DECIMAL
409 000516 010067 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
410 000522 000207 RETURN
411
412 000001 .END

```

```

228 .SBTIL SETFLG - SETUP THE FLAGS WORD
229
230 *** - SETFLG - SETUP THE FLAGS WORD
231
232 INPUT:
233 TEMP = START ADDRESS OF THE TEMPLATE
234
235 OUTPUT:
236 THE FLAGS WORD BIT DEFINITIONS ARE UPDATED
237
238
239
240 SETFLG:
241 000170 MOV R5, -(SP) ; SAVE R5
242 000170 MOV TEMP, R5 ; GET START OF TEMPLATE
243 000172 016705 MOV $OBUF, R0 ; POINT R0 AT CURRENT POSITION IN OUTPUT BUFFER
244 000202 012701 000002' MOV #ZDLC, R1 ; POINT R1 AT INPUT STRING
245 000206 012702 000007' MOV #ZDLC, R2 ; ..AND LENGTH OF STRING
246 000212 004767 000000G CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
247 000216 032765 000000G 000006 BIT #ZF.TIM, C.FLG(R5) ; TIMER SUPPORT NEEDED ?
248 000224 001406 BEQ 10$; IF EQ, NO
249 000226 012701 000000G MOV #SZTIM, R1 ; POINT R1 AT INPUT STRING
250 000232 012702 000000G MOV #SZTIML, R2 ; ..AND LENGTH OF STRING
251 000236 004767 000000G CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
252 000242 032765 000000G 000006 10$: BIT #ZF.MFL, C.FLG(R5) ; MARK FOR LOAD ?
253 000250 001406 BEQ 20$; IF EQ, NO .. LEAVE
254 000252 012701 000000G MOV #SZMFL, R1 ; POINT AT INPUT STRING
255 000256 012702 000000G MOV #SZMFL, R2 ; ..AND LENGTH
256 000262 004767 000000G CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
257 000266 032765 000000G 000006 20$: BIT #ZF.COUL, C.FLG(R5) ; DLC SUPPORT COUNTERS ?
258 000274 001406 BEQ 30$; IF EQ, NO
259 000276 012701 000000G MOV #SZCOUL, R1 ; POINT R1 AT INPUT STRING
260 000302 012702 000000G MOV #SZCOUL, R2 ; ..AND LENGTH OF STRING
261 000306 004767 000000G CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
262 000312 032765 000000G 000006 30$: BIT #ZF.MAN, C.FLG(R5) ; SUPPORT FOR NETWORK MANAGEMENT REQUESTS ?
263 000320 001406 BEQ 40$; IF EQ, NO
264 000322 012701 000011' MOV #ZMAN, R1 ; POINT R1 AT INPUT STRING
265 000326 012702 000007' MOV #ZMANL, R2 ; ..AND LENGTH OF STRING
266 000332 004767 000000G CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
267 000336 105740 40$: TSTB -(R0) ; BACK UP OVER LAST
268 000340 010067 000000G MOV R0, $OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
269 000344 012605 MOV (SP)+, R5 ; RESTORE R5
270 000346 000207 RETURN

```



```

LOCAL DATA
131
132
133
134
135
136 000000 000000 .SBTTL LOCAL DATA
137
138 000000
139 000002
140 000004
141 000016
142 000020
143 000022
144 000026
145 000000
146
147 000000 011 104 123 DSADF:
148 000010
149
150
151

; DEFINE THE DSA DESCRIPTOR BLOCK
;
; .ASECT
; =0
; .BLKW 1 ; C.LNK
; .BLKW 1 ; C.STS
; C.NUM: .BLKW 5 ; REMOTE DTE NUMBER (1.-15. DECIMAL DIGITS)
; C.SLO: .BLKW 1 ; SUBADDRESS LOW RANGE
; C.SHI: .BLKW 1 ; SUBADDRESS HI RANGE
; C.CUG: .BLKW 2 ; USER GROUP NAME
; C.LEN= . ; LENGTH OF THE BLOCK
;
; .PSECT
; .NLIST BEX
; .ASCII <11>/DSA$DF/<11>
; DSALN=-DSADF
; .LIST BEX
; .EVEN

```

UPDDSC      CREATED BY    MACRO    ON 29-JUN-85 AT 05:51      PAGE 1      F 6

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE       | REFERENCES                     |
|---------|-------------|--------------------------------|
| CS.MSK  | = ***** GX  | 6-174                          |
| CS.VAL  | = ***** GX  | 6-175                          |
| C.LEN   | = 000060    | #5-142                         |
| C.MSK   | 000004      | #5-140      6-174              |
| C.VAL   | 000032      | #5-141      6-175              |
| DSCDF   | 000000 R    | #5-145      5-146      6-170   |
| DSCLN   | = 000010    | #5-146      6-171              |
| F.NRBD  | = ***** GX  | *6-164      *6-178      *6-180 |
| \$BUFI  | = ***** GX  | 6-180                          |
| \$MVASC | = ***** GX  | 6-172                          |
| \$DBUF  | = ***** GX  | *6-173      6-176              |
| \$OFDB  | = ***** GX  | *6-164      *6-178      *6-180 |
| \$PTOPT | = ***** GX  | 6-174      6-175               |
| \$PTR50 | = ***** GX  | 6-174      6-175               |
| \$UPDSC | = 000010 RG | #6-163                         |
| \$WDATA | = ***** GX  | 6-179                          |
| .BUFF   | = ***** GX  | 6-164      6-169      6-177    |
| .COMMA  | = ***** GX  | *6-168                         |
| .FLAGS  | = ***** GX  | *6-167                         |
| .TEMP   | = ***** GX  | *6-165                         |

UPDDTE - UPDATE LOCAL DTE DESCR MACRO V05.03b Saturday 29-Jun-85 <sup>6 7</sup>05:51  
Table of contents

4- 57 MACRO CALLS AND LOCAL DATA  
4- 58 MACRO DEFINITIONS  
5- 135 LOCAL DATA  
6- 165 \$UPDTE - UPDATE DTE DEFINITION  
7- 202 SETSTA - SET STATE  
7- 203 SETLIN - SET LINE ID

```
56 .SBTTL MACRO DEFINITIONS
57
58 ;****
59 ; LOCAL MACROS
60 ;****
61
62 .MACRO PRINT TEXT
63 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM PRINT
66
67 ;****
68 ; MACRO CALLS
69 ;****
70
71 .MCALL ISTAT$,STATE$,TRANS$,FLTDF$
72 FLTDF$; DEFINE FILTER BLOCK SYMBOLS
73 000000
```

UPDFIL - CFE UPDATE REMOTE CHAR MACRO V05.03b Saturday 29-Jun-85 05:52<sup>6 9</sup>  
Table of contents

4- 46 MACRO CALLS AND LOCAL DATA  
5- 90 \$UPFIL - SETUP REMOTE DEFINITION  
6- 141 PUTFIL - WRITE OUT FILE SPEC

\$UPFEA - UPDATE FEATURES DEFINITION

```

126 .SBTTL $UPFEA - UPDATE FEATURES DEFINITION
127 ;+
128 *** - $UPFEA - UPDATE FEATURES DEFINITION
129
130 INPUT:
131 R3-R5 - TPARS REGISTERS
132
133 OUTPUT:
134 C-BIT = SUCCESS/FAILURE
135
136 ; -
137
138 000002 $UPFEA::
139 000002 012700 000000G 10$: MOV # $TMLST, R0 ; GET TEMPLATE LISTHEAD POINTER
140 000006 010001 MOV R0, R1 ; SAVE ADDRESS OF CURRENT TEMPLATE
141 000010 011000 MOV (R0), R0 ; GET NEXT TEMPLATE
142 000012 001436 BEQ 101$; IF EQ, END OF TEMPLATES
143 000014 122760 000000G 000000G CMPB #CS.FEA, C.STS(R0) ; IS THIS A FEATURES TEMPLATE ?
144 000022 001371 BNE 10$; IF NE, NO .. KEEP LOOKING
145
146 ; ASSUME THIS IS THE RIGHT TEMPLATE
147
148 000024 011011 MOV (R0), (R1) ; REMOVE TEMPLATE FROM CHAIN
149 000026 010067 177746 MOV R0, TEMP ; SAVE TEMPLATE ADDRESS
150 000032 032760 000000G 000000G BIT #CS.MOD, C.STS(R0) ; WAS THIS TEMPLATE MODIFIED ?
151 000040 001004 BNE 15$; IF NE, YES .. UPDATE THE DEFINITION
152 000042 012767 000000G 000000G MOV # $BUFI, $OFDB+F.NRBD+2 ; ELSE, OUTPUT THE INPUT BUFFER
153 000050 000416 BR 30$; AND LEAVE
154 000052 004767 000000G 15$: CALL $MOVST ; MOVE FIRST FIELD INTO OUTPUT BUFFER
155 000056 010546 MOV R5, -(SP) ; SAVE R5
156 000060 005001 CLR R1 ; IGNORE BLANKS
157 000062 012702 000000' MOV #FEAKTB, R2 ; KEYWORD TABLE
158 000066 012705 000000' MOV #FEASTB, R5 ; STATE TABLE
159 000072 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
160 000076 103002 BCC 20$; NORMAL RETURN IF NO ERROR
161 000100 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
162 000104 012605 MOV (SP)+, R5 ; RESTORE R5
163 000106 000207 20$: 30$: RETURN
164
165 ; ERROR CONDITIONS
166
167 000110 000167 000000G 101$: JMP .CONER ; CONSISTENCY ERROR

```

\$UPFEA - UPDATE FEATURES DEFINITION

SETNAM - CHECK IF LLC PROCESS NAMES MATCH

```

226 .SBTTL SETNAM - CHECK IF LLC PROCESS NAMES MATCH
227
228 *** - SETNAM - CHECK IF LLC PROCESS NAMES MATCH
229
230 INPUT:
231 .PSTPT = ADDRESS OF THE LLC PROCESS NAME
232 .PSTCN = LENGTH OF THE LLC PROCESS NAME
233
234 OUTPUT:
235 C-BIT = SUCCESS/FAILURE
236
237
238
239 000166 SETNAM: CALL $MOVPT ; MOV THE PROCESS NAME INTO THE OUTPUT BUFFER
240 000166 004767 000000G MOV .PSTPT,RO ; GET ADDRESS OF ASCII LLC PROCESS NAME
241 000172 016700 000000G MOV PC,R1 ; PERIODS ARE ACCEPTABLE RAD50 CHARS
242 000176 010701 CALL $CAT5 ; CONVERT PROCESS NAME TO RADRO
243 000200 004767 000000G MOV TEMP,RO ; POINT RO AT TEMPLATE
244 000204 016700 177570 CMP R1,C.NAM(RO) ; IS THIS THE RIGHT TEMPLATE ?
245 000210 020160 000004 BEQ 10$; IF EQ, YES .. OKAY
246 000214 001402 ADD #2,(SP) ; ELSE REJECT THE TRANSITION
247 000216 062716 000002 10$: RETURN
248 000222 000207

```

SETFLG - SETUP THE FLAGS WORD

.TITLE UPDLOG - CFE UPDATE LOGGING STATE DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

CFE - UPDATE STATE TABLE TO PARSE THE LOGGING STATE DEFINITION

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 07-JUL-80  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0



```

96 ;****
97 ; INITIALIZE TABLE GENERATION
98 ;****
99
100
101 000002 ISTAT$ NODSTB,NODKTB
102
103
104 000002 STATES$
105 000002 TRANS$ LA,, $MOVPT
106
107 000002 STATES$
108 000002 TRANS$ $RAD50,,SETNAM
109
110 000002 STATES$
111 000002 TRANS$ RA,, $MOVPT
112
113 000002 STATES$
114 000002 TRANS$ <','>,, $MOVPT
115
116 000002 STATES$
117 000002 TRANS$ LA,, $MOVPT
118
119 000002 STATES$
120 000002 TRANS$!ID,,SETNID
121
122 000002 STATES$
123 000002 TRANS$ <','>,, $MOVPT
124
125 000002 STATES$
126 000002 TRANS$ $NUMBR,ADD2,SETNNA
127 000002 TRANS$ $LAMDA,2,SETNNA
128
129 000002 STATES$
130 000002 TRANS$ ADD2
131 000002 TRANS$ <','>,, $MOVPT
132
133 000002 STATES$
134 000002 TRANS$ $NUMBR,,SETNNM
135
136 000002 STATES$
137 000002 TRANS$ <','>,, $MOVPT
138
139 000002 STATES$
140 000002 TRANS$ $NUMBR,HOS2,SETHOA
141 000002 TRANS$ $LAMDA,,SETHOA
142
143 000002 STATES$
144 000002 TRANS$ HOS2
145 000002 TRANS$ <','>,, $MOVPT
146
147 000002 STATES$
148 000002 TRANS$ $NUMBR,,SETHOS
149
150 ;
151 ; EXIT STATE
152 ;
 STATES$ EXIT
 TRANS$ $LAMDA,$EXIT,$CLNUP

```

```

55 ;****
56 ; LOCAL MACROS
57 ;****
58
59 .MACRO PRINT TEXT
60 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
61 CALL $PRINT ; PRINT MESSAGE
62 .ENDM PRINT
63
64 ;****
65 ; MACRO CALLS
66 ;****
67
68 .MCALL ISTAT$,STATE$,TRANS
69
70 ;****
71 ; LOCAL DATA
72 ;****
73
74 000074 LA = '< ; LEFT ANGLE BRACKET
75 000076 RA = '> ; RIGHT ANGLE BRACKET
76 000000 TEMP: .BLKW 1 ; ADDRESS OF OBJECT TEMPLATE
77 000002 060 054 124 OBJDEF: .ASCIZ /0,TASK,0,0,0/
78 000005 101 123 113
79 000010 054 060 054
80 000013 060 054 060
81 000016 000
82
83 78 000014 OBJLEN=<.-OBJDEF>-1
84 79 000017 011 117 102 OBJDF: .ASCII <11>/OBJ$DF/<11>
85 000022 112 044 104
86 000025 106 011
87
88 80 000010 OBJLN=.-OBJDF
89 81 000027 .BLKB 80. ; OUTPUT BUFFER
90 82 .EVEN
91
92 ;
93 ; DEFINE OBJECT TEMPLATE
94 ;
95 000000 .ASECT
96 000000 000000 .=0
97 000000 .BLKW 1 ; C.LNK
98 000002 .BLKW 1 ; C.STS
99 000004 C.TYP: .BLKB 1 ; OBJECT TYPE
100 000005 C.FLG: .BLKB 1 ; OBJECT FLAG
101 000006 C.NAM: .BLKW 2 ; OBJECT TASK NAME (RAD50)
102 000012 C.CPY: .BLKB 1 ; MAXIMUM NUMBER OF COPIES OF A TASK TO SPAWN
103
104 94 000013 C.LEN=.
105 95 000150 .PSECT
106 96
107 97

```

UPDPAR - CFE UPDATE PARTITION D MACRO V05.03b Saturday 29-Jun-85 05:54 F 15  
Table of contents

|     |     |                                       |
|-----|-----|---------------------------------------|
| 6-  | 147 | \$UPPAR - UPDATE PARTITION DEFINITION |
| 7-  | 186 | SETNMO - SET UP PARTITION NAME 0      |
| 8-  | 214 | SETNM1 - SET UP PARTITION NAME 1      |
| 9-  | 240 | SETALL - SET THE BYTE-AREA EXTENSION  |
| 10- | 275 | SETTOP - SETUP TOP FLAG               |
| 11- | 299 | SETSBP - SETUP SBPOOL FLAG            |

UPDPAR      CREATED BY MACRO ON 29-JUN-85 AT 05:54      PAGE 2      F 16  
 MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGIPS     | #5-90      | #5-94  | #5-110 | #5-114 | #5-118 | #5-122 | #5-123 | #5-128 | #5-135 |        |
| ISTATS     | #4-58      | 5-90   |        |        |        |        |        |        |        |        |
| MTRANS     | #5-90      |        |        |        |        |        |        |        |        |        |
| STATS      | #4-58      | 5-92   | #5-96  | #5-99  | #5-102 | #5-105 | #5-108 | #5-112 | #5-116 | #5-120 |
|            | #5-126     | #5-130 | #5-133 | #5-137 | #5-142 | #5-145 |        |        |        |        |
| TRANS      | #4-58      | #5-93  | #5-94  | #5-97  | #5-100 | #5-103 | #5-106 | #5-109 | #5-110 | #5-113 |
|            | #5-114     | #5-117 | #5-118 | #5-121 | #5-122 | #5-123 | #5-127 | #5-128 | #5-131 | #5-134 |
|            | #5-135     | #5-138 | #5-143 |        |        |        |        |        |        |        |

SETFLG - SETUP THE FLAGS WORD

```

255 .SBTTL SETFLG - SETUP THE FLAGS WORD
256 ;+
257 ;*** - SETFLG - SETUP THE FLAGS WORD
258
259 INPUT:
260 TEMP = START ADDRESS OF THE TEMPLATE
261
262 OUTPUT:
263 THE FLAGS WORD BIT DEFINITIONS ARE UPDATED.
264
265 :-
266
267 000252 SETFLG:
268 000252 MOV R5, -(SP) ; SAVE R5
269 000254 MOV TEMP, R5 ; GET START OF TEMPLATE
270 000260 MOV $OBUF, R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
271 000264 MOV #ZDDM, R1 ; POINT R1 AT INPUT STRING
272 000270 MOV #ZDDML, R2 ; ..AND LENGTH OF STRING
273 000274 CALL $MVASC ; MARK PROCESS AS A DDM
274 000300 PUTFLG TIM, <$> ; TIMER BIT NEEDED ?
275 000324 PUTFLG MFL, <$> ; MARK FOR LOAD ?
276 000350 PUTFLG DLC ; IS THIS A DLC ?
277 000374 PUTFLG COU, <$> ; COUNTERS SUPPORTED ?
278 000420 PUTFLG KMX ; KMC DEVICE ?
279 000444 PUTFLG LMC ; LOAD MICRO-CODE ?
280 000470 PUTFLG MUX ; IS THIS A MUX DEVICE ?
281 000514 PUTFLG PSE ; PSEUDO SLT FOR THIS DEVICE ?
282 000540 PUTFLG DIA ; DOES DEVICE REQUIRE DIAGNOSTIC u-CODE ?
283 000564 PUTFLG DVP ; DOES PROCESS RUN AT DEVICE PRIORITY ?
284 000610 PUTFLG MAN ; DOES PROCESS HAVE NET. MAN. ENTRY POINT ?
285 000634 TSTB -(R0) ; BACK UP OVER LAST '1'
286 000636 MOV R0, $OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
287 000642 MOV (SP)+, R5 ; RESTORE R5
288 000644 RETURN

```

SETPRI - SET THE PRIORITY

|     |        |        |                      |                          |
|-----|--------|--------|----------------------|--------------------------|
| 118 | 000002 | TRANS  | \$NUMBR,,SETLST      | ; LINK SERVICE THRESHOLD |
| 119 |        | STATES |                      |                          |
| 120 | 000002 | TRANS  | <','>,, \$MOVPT      |                          |
| 121 | 000002 |        |                      |                          |
| 122 |        | STATES |                      |                          |
| 123 | 000002 | TRANS  | \$NUMBR,,SETRET      | ; RETRANSMIT FACTOR      |
| 124 | 000002 |        |                      |                          |
| 125 |        | STATES |                      |                          |
| 126 | 000002 | TRANS  | <','>,, \$MOVPT      |                          |
| 127 | 000002 |        |                      |                          |
| 128 |        | STATES |                      |                          |
| 129 | 000002 | TRANS  | \$NUMBR,,SETSEG      | ; SEGMENT SIZE           |
| 130 | 000002 |        |                      |                          |
| 131 |        |        |                      |                          |
| 132 |        |        |                      |                          |
| 133 |        |        |                      |                          |
| 134 | 000002 | STATES | EXIT                 |                          |
| 135 | 000002 | TRANS  | \$EOS,\$EXIT,\$CLNUP |                          |
| 136 |        |        |                      |                          |
| 137 | 000002 | STATES |                      |                          |

; EXIT STATE

|                  |                     |                     |                   |                   |
|------------------|---------------------|---------------------|-------------------|-------------------|
| CD.LEN= 000022   | DECSTB 000000RG     | 002 NT\$INT= 000004 | SETQUT 000220R    | \$LAMDA= 000000   |
| CS.DEC= ***** GX | EXIT 000124R        | 002 NT\$RET= 000032 | SETRET 000440R    | \$MOVPT= ***** GX |
| CS.MOD= ***** GX | F.NRBD= ***** GX    | NT\$ROU= 000024     | SETSEG 000470R    | \$MOVST= ***** GX |
| C.DELF 000016    | NM\$ARA= 176000     | NT\$RTR= 000030     | TEMP 000000R      | \$NUMBR= 000002   |
| C.DELW 000017    | NM\$NOD= 001777     | NT\$TSP= 000026     | \$ALPHA= 000022   | \$OBUF = ***** GX |
| C.INAC 000014    | NT\$AKD= 000020     | N\$SACK= 000011     | \$ANY = 000020    | \$OFDB = ***** GX |
| C.INCT 000012    | NT\$AKI= 000022     | N\$SHDR= 000007     | \$BLANK= 000006   | \$RAD50= 000016   |
| C.IPL 000020     | NT\$CC = 000016     | N\$SOVR= 000022     | \$BUFI = ***** GX | \$STRNG= 000004   |
| C.LKS 000010     | NT\$CON= 000000     | SETDEF 000300R      | \$CLNUP= ***** GX | \$SUBXP= 000010   |
| C.LST 000021     | NT\$CTL= 000000     | SETDEW 000330R      | \$CNVD = ***** GX | \$TMLST= ***** GX |
| C.NDC 000006     | NT\$DAT= 000002     | SETINA 000250R      | \$DIGIT= 000024   | \$UPDEC 000002RG  |
| C.QUIT 000013    | NT\$DC = 000012     | SETINC 000170R      | \$DNUMB= 000014   | \$\$\$FLG= 177777 |
| C.RETF 000015    | NT\$DIS= 000014     | SETIPL 000360R      | \$EOS = 000012    | \$\$\$KEY= 177777 |
| C.SEG 000004     | NT\$DLS= 000006     | SETLKS 000114R      | \$ERROR= ***** GX | \$\$\$STA= 000000 |
| C.STS = ***** GX | NT\$ILS= 000010     | SETLST 000410R      | \$EXIT = 000000   | .CONER= ***** GX  |
| DECKTB 000000RG  | 003 NT\$IMS= 000002 | SETNDC 000142R      | \$FAIL = 177777   | .TPARS= ***** GX  |

. ABS. 000022 000 (RW,I,GBL,ABS,OVR)  
 000524 001 (RW,I,LCL,REL,CON)  
 \$STATE 000132 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000000 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000000 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10675 Words ( 42 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:34.31  
 SY:UPDDEC.V2,[132,134]UPDDEC/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDDEC

SETPRI - SET THE PRIORITY

```

272 .SBTTL SETPRI - SET THE PRIORITY
273 ;+
274 *** - SETPRI - SET THE PROCESS PRIORITY
275 ;
276 INPUT:
277 .PNUMB = PRIORITY
278 .PNUMH = PRIORITY (HIGH ORDER)
279 ;
280 OUTPUT:
281 ;
282 ;
283 ;
284 SETPRI:
285 MOV TEMP,R2 ; GET START OF TEMPLATE
286 MOV C.PRI(R2),R1 ; GET THE PROCESS PRIORITY
287 MOV $0BUF,R0 ; GET THE CURRENT POSITION IN OUTPUT BUFFER
288 CALL $CNVO ; CONVERT BINARY TO OCTAL
289 MOV R0,$0BUF ; UPDATE THE POINTER INTO THE OUTPUT BUFFER
290 RETURN

```

SETNLT - STORE NUMBER OF LINE TABLES



```

153 .SBTTL $UPDSA - UPDATE PERMANENT VIRTUAL CIRCUIT
154 ;+
155 ; *** - $UPDSA - UPDATE PERMANENT VIRTUAL CIRCUIT
156 ;
157 INPUT:
158 R5 - TEMPLATE ADDRESS
159 ;
160 OUTPUT:
161 C-BIT = SUCCESS/FAILURE
162 ;
163 ; -
164 ;
165 000010 $UPDSA::
166 000010 012767 000000G 000000C MOV #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
167 000016 010567 000000G MOV R5,.TEMP ; SAVE TEMPLATE ADDRESS
168 000022 005725 TST (R5)+ ; POINT TO C.STS
169 000024 010567 000000G MOV R5,.FLAGS ; SAVE ADDRESS OF FLAGS WORD
170 000030 012767 177777 000000G MOV #-1,.COMMA ; INITIALIZE FOR OPTIONS
171 000036 012700 000000G MOV #.BUFF,R0 ; POINT AT START OF OUTPUT BUFFER
172 000042 012701 000000' MOV #DSADF,R1 ; POINT AT START OF DSA DEFINITION
173 000046 012702 000010 MOV #DSALN,R2 ; GET LENGTH OF DSA DEFINITION
174 000052 004767 000000G CALL $MVASC ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
175 000056 010067 000000G MOV R0,$OBUF ; SAVE CURRENT POSITION IN OUTPUT BUFFER
176 000062 $PTR50 #C.NUM,#5,#CS.NUM ; REMOTE DTE NUMBER
177 000110 $PTNUM #C.SLO,#CS.SLO ; SUBADDRESS HIGH
178 000134 $PTNUM #C.SHI,#CS.SHI ; SUBADDRESS LOW
179 000160 $PTR50 #C.CUG,#2,#DA$CUG ; GROUP NAME
180 000206 016701 000000G MOV $OBUF,R1 ; CALCULATE LENGTH OF BUFFER
181 000212 162701 000000G SUB #.BUFF,R1 ;
182 000216 010167 000000C MOV R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
183 000222 004767 000000G CALL $WDATA ; WRITE THE RECORD TO THE OUTPUT FILE
184 000226 012767 000000G 000000C MOV #SBUF1,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE 'END$DF' LINE
185 000234 000207 RETURN
186
187 000001 .END

```

UPDDSC      CREATED BY MACRO ON 29-JUN-85 AT 05:51      PAGE 2      G 6  
MACRO CROSS REFERENCE      CREF    04.00

| MACRO NAME | REFERENCES                   |
|------------|------------------------------|
| EPRINT     | #4-79                        |
| ERRORS     | #4-68                        |
| PRINT      | #4-63                        |
| \$PTNMB    | #4-112                       |
| \$PTNUM    | #4-111                       |
| \$PTR50    | #4-116      6-174      6-175 |
| \$\$\$PTN  | #4-91      4-111      4-112  |

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55

.TITLE UPDDTE - UPDATE LOCAL DTE DESCRIPTOR IN CETAB  
.IDENT /V05.00/  
.ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

UPDATE LOCAL DTE DESCRIPTOR IN CETAB

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-AUG-81  
DECNET-11M/S V3.1  
DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

LOCAL DATA

```

75 .SBTTL LOCAL DATA
76
77 ;****
78 ; LOCAL DATA
79 ;****
80
81 000074 LA = '<' ; LEFT ANGLE BRACKET
82 000076 RA = '>' ; RIGHT ANGLE BRACKET
83 000000 TEMP: .BLKW 1 ; ADDRESS OF EVENT TEMPLATE
84 000002 EVTOFF: .BLKW 1 ; OFFSET INTO EVENTS
85 000004 060 054 074 EVTDEF: .ASCIZ /0,<0,0,0>,.CONSOLE/
86 000023 011 105 126 EVTLEN=<.-EVTDEF>-1
87 000030 000010 EVTDF: .ASCII <11>/EVT$DF/<11>
88 000040 115 117 116 EVTLN=.-EVTDF
89 000047 106 111 114 MONITR: .ASCII /MONITOR/
90 000053 103 117 116 MONL=.-MONITR
91 000062 000007 FILE: .ASCII /FILE/
92 000076 000004 FILEL=.-FILE
93 000080 000004 CONSLE: .ASCII /CONSOLE/
94 000084 000007 CONL=.-CONSLE
95 000088 80. ; OUTPUT BUFFER
96 000092 BUFF: .BLKB 80.
97 000096 .EVEN
98 ;
99 ; DEFINE EVENT TEMPLATE
100 000040 .ASECT
101 000000 =0
102 000000 .BLKW 1 ; C.LNK
103 000002 .BLKW 1 ; C.STS
104 000004 C.CLS: .BLKW 1 ; EVENT CLASS
105 000006 C.EVT: .BLKW 4 ; EVENT MASKS
106 000016 C.SNK: .BLKW 1 ; SINK TYPE
107 000020 C.LEN=.
108 000202 .PSECT
109

```

STATE TABLES

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44

.TITLE UPDFIL - CFE UPDATE REMOTE CHARACTERISTICS IN CETAB  
.IDENT /V05.00/

.....  
COPYRIGHT (C) 1983, 1984, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

.....  
THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

.....  
THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

.....  
DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

.....  
MODULE DESCRIPTION:

STATE TABLE TO PARSE THE REMOTE DEFINITIONS

.....  
DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

.....  
IDENT HISTORY:

.....  
4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0

.....  
5.00 22-JUL-85  
DECnet-11M/s V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RX V1.0  
.....

```

169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184 000114
185 000114 016700 000000G
186 000120 016702 177654
187 000124 016201 000012
188 000130 004767 000000G
189 000134 010067 000000G
190 000140 000207
191
192
193 000001

 .SBTTL SETWD3 - UPDATE THE CURRENT STATES WORD
 ;+
 ;*** - SETWD3 - UPDATE THE CURRENT STATES WORD
 ;INPUT:
 ;TEMP = THE TEMPLATE ADDRESS
 ;C.WD3(TEMP) = THE CURRENT STATES WORD
 ;$OBUF = THE CURRENT POSITION IN THE OUTPUT BUFFER
 ;OUTPUT:
 ;$OBUF = THE NEW POSITION IN THE OUTPUT BUFFER
 ;-
 SETWD3:
 MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
 MOV TEMP,R2 ; POINT AT THE TEMPLATE
 MOV C.WD3(R2),R1 ; GET THE CURRENT STATES WORD
 CALL $CNVD ; CONVERT THE NUMBER TO ASCII
 MOV R0,$OBUF ; UPDATE THE POSITION IN THE OUTPUT BUFFER
 RETURN
 .END

```

```

250 .SBTTL SETFLG - SETUP THE FLAGS WORD
251
252 *
253 *** - SETFLG - SETUP THE FLAGS WORD
254
255 INPUT:
256 TEMP = START ADDRESS OF THE TEMPLATE
257
258 OUTPUT:
259 THE FLAGS WORD BIT DEFINITIONS ARE UPDATED
260
261 -
262 SETFLG:
263 MOV R5, -(SP) ; SAVE R5
264 MOV TEMP, R5 ; GET START OF TEMPLATE
265 MOV $OBUF, R0 ; POINT AT CURRENT IN OUTPUT BUFFER
266 MOV #ZLLC, R1 ; POINT R1 AT INPUT STRING
267 MOV #ZLLCL, R2 ; ..AND LENGTH OF STRING
268 CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
269 BIT #ZF.TIM, C.FLG(R5) ; TIMER BIT NEEDED ?
270 BEQ 10$; IF EQ, NO
271 MOV #ZTIM, R1 ; POINT R1 AT INPUT STRING
272 MOV #ZTIML, R2 ; ..AND LENGTH OF STRING
273 CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
274 BIT #ZF.MFL, C.FLG(R5) ; MARK FOR LOAD ?
275 BEQ 20$; IF EQ, NO
276 MOV #ZMFL, R1 ; POINT AT INPUT STRING
277 MOV #ZMFL, R2 ; ..AND LENGTH OF STRING
278 CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
279 BIT #ZF.MTM, C.FLG(R5) ; MULTIPLE TIMERS ?
280 BEQ 30$; IF EQ, NO
281 MOV #ZMTM, R1 ; POINT AT INPUT STRING
282 MOV #ZMTML, R2 ; ..AND LENGTH OF STRING
283 CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
284 BIT #ZF.COU, C.FLG(R5) ; SUPPORT FOR COUNTERS ?
285 BEQ 40$; IF EQ, NO .. LEAVE
286 MOV #ZCOU, R1 ; POINT AT INPUT STRING
287 MOV #ZCOUL, R2 ; ..AND LENGTH OF STRING
288 CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
289 BIT #ZF.X3P, C.FLG(R5) ; SUPPORT FOR X.25 LEVEL 3?
290 BEQ 50$; IF EQ, NO .. LEAVE
291 MOV #ZX3P, R1 ; POINT AT INPUT STRING
292 MOV #ZX3PL, R2 ; ..AND LENGTH OF STRING
293 CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
294 BIT #ZF.SLI, C.FLG(R5) ; DOES IT USE THE SYSTEM LEVEL INTERFACE ?
295 BEQ 55$; IF EQ, NO .. LEAVE
296 MOV #ZSLI, R1 ; POINT AT INPUT STRING
297 MOV #ZSLIL, R2 ; ..AND LENGTH OF STRING
298 CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
299 BIT #ZF.INI, C.FLG(R5) ; DOES IT REQUIRE INIT/EXIT SUPPORT?
300 BEQ 60$; IF EQ, NO .. LEAVE
301 MOV #ZINI, R1 ; POINT AT INPUT STRING
302 MOV #ZINIL, R2 ; ..AND LENGTH OF STRING
303 CALL $MVASC ; MOVE STRING INTO OUTPUT BUFFER
304 TSTB -(R0) ; BACK UP OVER LAST ""
305 MOV R0, $OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
306 MOV (SP)+, R5 ; RESTORE R5
307 RETURN

```

```

55 .SBTTL MACRO DEFINITIONS
56
57 ;****
58 ; LOCAL MACROS
59 ;****
60
61 .MACRO PRINT TEXT
62 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
63 CALL $PRINT ; PRINT MESSAGE
64 .ENDM PRINT
65
66 .MACRO ERROR$ TEXT
67 ;
68 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
69 ; MESSAGE STRING.
70 ;
71 .IF DIF <TEXT><RO>
72 MOV TEXT,RO ; GET ADDRESS OF ERROR MESSAGE
73 .ENDC
74 CALL $CFERR ; PRINT ERROR MESSAGE
75 .ENDM ERROR$
76
77 .MACRO EPRINT TEXT
78 ;
79 ; PRINT TEXT ON ERROR LUN
80 ;
81 .IF DIF <TEXT><RO>
82 MOV TEXT,RO ; GET ADDRESS OF ASCIZ STRING TO PRINT
83 .ENDC
84 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
85 .ENDM EPRINT
86
87 ;****
88 ; MACRO CALLS
89 ;****
90 .MCALL ISTAT$,STATE$,TRANS$

```



153 000002  
154 000002  
155 000002  
156  
157 000002

STATES ID  
TRANS RA,\$EXIT  
TRANS \$ANY,ID  
STATES

```

99 ;****
100 ; INITIALIZE TABLE GENERATION
101 ;****
102
103
104 000150 ISTAT$ OBJSTB,OBJKTB
105
106
107 000150 STATES$
108 000150 TRANS$ $NUMBR,,SETYPE
109
110 000150 STATES$
111 000150 TRANS$ <','>,$MOVPT
112
113 000150 STATES$
114 000150 TRANS$ $RAD50,,SETNAM
115
116 000150 STATES$
117 000150 TRANS$ <','>,$MOVPT
118
119 000150 STATES$
120 000150 TRANS$ $NUMBR,,SETVfy
121
122 000150 STATES$
123 000150 TRANS$ <','>,$MOVPT
124
125 000150 STATES$
126 000150 TRANS$ $NUMBR,,SETFLG
127
128 000150 STATES$
129 000150 TRANS$ <','>,$MOVPT
130
131 000150 STATES$
132 000150 TRANS$ $NUMBR,,SETCPY
133
134 ; EXIT STATE
135 ;
136 000150 STATES$ EXIT
137 000150 TRANS$ $EOS,$EXIT
138
139 000150 STATES$

```

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51

.TITLE UPDPAR - CFE UPDATE PARTITION DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1984, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE PARTITION DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/s V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RXS V1.0

```

UU UU P P P P P P P P D D D D D D D D P P P P P P P P S S S S S S S S N N N N
UU UU P P P P P P P P D D D D D D D D P P P P P P P P S S S S S S S S N N N N
UU UU P P P P D D D D P P P P S S S S N N N N
UU UU P P P P D D D D P P P P S S S S N N N N
UU UU P P P P D D D D P P P P S S S S N N N N N N
UU UU P P P P D D D D P P P P S S S S N N N N N N
UU UU P P P P P P P P D D D D P P P P P P P P S S S S S S N N N N N N
UU UU P P P P P P P P D D D D P P P P P P P P S S S S S S N N N N N N
UU UU P P D D D D D D P P P P S S S S N N N N N N
UU UU P P D D D D D D P P P P S S S S N N N N
UU UU P P D D D D D D P P P P S S S S N N N N
UU UU P P D D D D D D P P P P S S S S N N N N
UUUUUUUUUU P P D D D D D D D D P P S S S S S S S S N N N N
UUUUUUUUUU P P D D D D D D D D P P S S S S S S S S N N N N

```

```

LL SSSSSSSS TTTT TTTT TTTT
LL SSSSSSSS TTTT TTTT TTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

```

290
291
292
293
294
295
296
297
298
299
300
301
302 000646
303 000646 016702 177126
304 000652 016201 000010
305 000656 016700 000000G
306 000662 004767 000000G
307 000666 010067 000000G
308 000672 000207

 .SBTTL SETPRI - SET THE PRIORITY
 ;+
 ;*** - SETPRI - SET THE PROCESS PRIORITY
 ;
 ;INPUT:
 ; .PNUMB = PRIORITY
 ; .PNUMH = PRIORITY (HIGH ORDER)
 ;
 ;OUTPUT:
 ;-
 ;
 SETPRI:
 MOV TEMP,R2 ; GET START OF TEMPLATE
 MOV C.PRI(R2),R1 ; GET PRIORITY
 MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
 CALL $CNVO ; CONVERT NUMBER TO OCTAL
 MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
 RETURN

```

```

139 .SBTTL $UPDEC - UPDATE DEC DEFINITION
140 +
141 *** - $UPDEC - UPDATE DEC$DF DEFINITION
142
143 INPUT:
144 R3-R5 - TPARS REGISTERS
145
146 OUTPUT:
147 C-BIT = SUCCESS/FAILURE
148
149 -
150
151 $UPDEC::
152 000002 012700 000000G MOV #STMLST,R0 ; POINT AT START OF TEMPLATES
153 000002 010001 10$: MOV R0,R1 ; SAVE PREVIOUS TEMPLATE ADDRESS
154 000010 011000 MOV (R0),R0 ; GET NEXT TEMPLATE
155 000012 001436 BEQ 101$; IF EQ, CANNOT FIND LLC TEMPLATE
156 000014 122760 000000G 000000G CMPB #CS.DEC,C.STS(R0) ; IS THIS AN DEC TEMPLATE ?
157 000022 001371 BNE 10$; IF NE, NO .. KEEP LOOKING
158
159 ; ASSUME THIS IS THE RIGHT TEMPLATE
160
161 000024 011011 MOV (R0),(R1) ; REMOVE TEMPLATE FROM CHAIN
162 000026 010067 177746 MOV R0,TEMP ; SAVE TEMPLATE ADDRESS
163 000032 032760 000000G 000000G BIT #CS.MOD,C.STS(R0) ; WAS THIS TEMPLATE MODIFIED ?
164 000040 001004 BNE 15$; IF NE, YES .. UPDATE THE RECORD
165 000042 012767 000000G 000000C MOV #BUF1,$OFDB+F.NRBD+2 ; ELSE, OUTPUT THE INPUT BUFFER
166 000050 000416 BR 30$; AND LEAVE
167 000052 004767 15$: CALL $MOVST ; MOVE FIRST FIELD INTO OUTPUT BUFFER
168 000056 010546 MOV R5,-(SP) ; SAVE R5
169 000060 005001 CLR R1 ; IGNORE BLANKS
170 000062 012702 000000' MOV #DECKTB,R2 ; KEYWORD TABLE
171 000066 012705 000000' MOV #DECSTB,R5 ; STATE TABLE
172 000072 004767 000000G CALL .TPARS ; PARSE THE REST OF THE LINE
173 000076 103002 000000G BCC 20$; NORMAL RETURN IF NO ERROR
174 000100 005267 INC $ERROR ; INDICATE SYNTAX ERROR
175 000104 012605 20$: MOV (SP)+,R5 ; RESTORE R5
176 000106 000207 30$: RETURN
177
178 ; ERROR CONDITIONS
179
180 000110 000167 000000G 101$: JMP .CONER ; CONSISTENCY ERROR

```

UPDDEC CREATED BY MACRO ON 29-JUN-85 AT 05:49 PAGE 1 H 3  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL   | VALUE      | REFERENCES                                                          |
|----------|------------|---------------------------------------------------------------------|
| CD.LEN   | = 000022   | #4-52                                                               |
| CS.DEC   | = ***** GX | 6-156                                                               |
| CS.MOD   | = ***** GX | 6-163                                                               |
| C.DELF   | 000016     | #4-52 12-300                                                        |
| C.DELW   | 000017     | #4-52 13-321                                                        |
| C.INAC   | 000014     | #4-52 11-279                                                        |
| C.INCT   | 000012     | #4-52 9-237                                                         |
| C.IPL    | 000020     | #4-52 14-342                                                        |
| C.LKS    | 000010     | #4-52 7-196                                                         |
| C.LST    | 000021     | #4-52 15-363                                                        |
| C.NDC    | 000006     | #4-52 8-216                                                         |
| C.QUIT   | 000013     | #4-52 10-258                                                        |
| C.RETF   | 000015     | #4-52 16-384                                                        |
| C.SEG    | 000004     | #4-52 17-404                                                        |
| C.STS    | = ***** GX | 6-156 6-163                                                         |
| DECTB    | 000000 RG  | #5-67 6-170                                                         |
| DECSTB   | 000000 RG  | #5-67 6-171                                                         |
| F.NRBD   | = ***** GX | *6-165                                                              |
| N\$SOVR  | = 000022   | 17-406                                                              |
| SETDEF   | 000300 R   | #12-297                                                             |
| SETDEW   | 000330 R   | #13-318                                                             |
| SETINA   | 000250 R   | #11-276                                                             |
| SETINC   | 000170 R   | #9-234                                                              |
| SETIPL   | 000360 R   | #14-339                                                             |
| SETLKS   | 000114 R   | #7-194                                                              |
| SETLST   | 000410 R   | #15-360                                                             |
| SETNDC   | 000142 R   | #8-214                                                              |
| SETOUT   | 000220 R   | #10-255                                                             |
| SETRET   | 000440 R   | #16-381                                                             |
| SETSEG   | 000470 R   | #17-402                                                             |
| TEMP     | 000000 R   | #4-59 *6-162 7-195 8-215 9-235 10-256 11-277 12-298 13-319          |
|          |            | 14-340 15-361 16-382 17-403                                         |
| \$ALPHA  | = 000022   | #5-67                                                               |
| \$ANY    | = 000020   | #5-67                                                               |
| \$BLANK  | = 000006   | #5-67                                                               |
| \$BUFI   | = ***** GX | 6-165                                                               |
| \$CNVD   | = ***** GX | 7-198 8-218 9-239 10-260 11-281 12-302 13-323 14-344 15-365         |
|          |            | 16-386 17-408                                                       |
| \$DIGIT  | = 000024   | #5-67                                                               |
| \$DNUMB  | = 000014   | #5-67                                                               |
| \$EOS    | = 000012   | #5-67                                                               |
| \$ERROR  | = ***** GX | *6-174                                                              |
| \$EXIT   | = 000000   | #5-67                                                               |
| \$FAIL   | = 177777   | #5-67                                                               |
| \$GPRM   | = *****    | 5-67                                                                |
| \$LAMDA  | = 000000   | #5-67                                                               |
| \$MOVST  | = ***** GX | 6-167                                                               |
| \$NUMBER | = 000002   | #5-67                                                               |
| \$OBUF   | = ***** GX | 7-197 *7-199 8-217 *8-219 9-238 *9-240 10-259 *10-261 11-280        |
|          |            | *11-282 12-301 *12-303 13-322 *13-324 14-343 *14-345 15-364 *15-366 |
|          |            | 16-385 *16-387 17-407 *17-409                                       |
| \$OFDB   | = ***** GX | *6-165                                                              |

```

292 .SBTTL SETNLT - STORE NUMBER OF LINE TABLES
293 :+
294 :*** - SETNLT - STORE NUMBER OF LINE TABLES
295 :
296 :INPUT:
297 :.PNUMB = NUMBER OF LINE TABLE TO BE SET IN PROCESS SPACE
298 :TEMP = TEMPLATE ADDRESS
299 :
300 :OUTPUT:
301 :C.NLT IN TEMPLATE MOVED INTO OUTPUT BUFFER
302 :
303 :-
304
305 000376 SETNLT:
306 000376 016702 177376 MOV TEMP,R2 ; GET THE TEMPLATE ADDRESS
307 000402 005001 CLR R1 ; GET THE LINE TABLE COUNT
308 000404 156201 000012 BISB C.NLT(R2),R1 ; ..WITHOUT SIGN EXTEND
309 000410 016700 000000G MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
310 000414 004767 000000G CALL $CNVD ; CONVERT NUMBER TO DECIMAL
311 000420 010067 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
312 000424 000207 RETUR

```



UPDDSA - UPDATE DESTINATION ADD MACRO V05.03b Saturday 29-Jun-85 05:50 Page 6-1  
Symbol table

|                  |                   |                   |                   |                   |
|------------------|-------------------|-------------------|-------------------|-------------------|
| CS.NUM= ***** GX | C.NUM 000004      | DSALN = 000010    | \$OFDB = ***** GX | \$WDATA= ***** GX |
| CS.SHI= ***** GX | C.SHI 000020      | F.NRBD= ***** GX  | \$PTNUM= ***** GX | .BUFF = ***** GX  |
| CS.SLO= ***** GX | C.SLO 000016      | \$BUFI = ***** GX | \$PTOPT= ***** GX | .COMMA= ***** GX  |
| C.CUG 000022     | DA\$CUG= ***** GX | \$MVASC= ***** GX | \$PTR50= ***** GX | .FLAGS= ***** GX  |
| C.LEN = 000026   | DSADF 000000R     | \$QBUF = ***** GX | \$UPDSA 000010RG  | .TEMP = ***** GX  |

. ABS. 000026 000 (RW,I,GBL,ABS,OVR)  
000236 001 (RW,I,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 8830 Words ( 35 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:05.47  
SY:UPDDSA.V2,[132,134]UPDDSA/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDDSA

\*\*FILE\*\*ID\*\*UPDDST

|           |    |          |          |          |          |           |      |
|-----------|----|----------|----------|----------|----------|-----------|------|
| UU        | UU | PPPPPPP  | DDDDDDDD | DDDDDDDD | SSSSSSSS | TTTTTTTTT |      |
| UU        | UU | PPPPPPP  | DDDDDDDD | DDDDDDDD | SSSSSSSS | TTTTTTTTT |      |
| UU        | UU | PP       | DD       | DD       | SS       | TT        |      |
| UU        | UU | PP       | DD       | DD       | SS       | TT        |      |
| UU        | UU | PP       | DD       | DD       | SS       | TT        |      |
| UU        | UU | PP       | DD       | DD       | SS       | TT        |      |
| UU        | UU | PPPPPPP  | DD       | DD       | SSSSSS   | TT        |      |
| UU        | UU | PPPPPPP  | DD       | DD       | SSSSSS   | TT        |      |
| UU        | UU | PP       | DD       | DD       | SS       | TT        |      |
| UU        | UU | PP       | DD       | DD       | SS       | TT        |      |
| UU        | UU | PP       | DD       | DD       | SS       | TT        | .... |
| UU        | UU | PP       | DD       | DD       | SS       | TT        | .... |
| UUUUUUUUU | PP | DDDDDDDD | DDDDDDDD | SSSSSSSS | TT       | ....      |      |
| UUUUUUUUU | PP | DDDDDDDD | DDDDDDDD | SSSSSSSS | TT       | ....      |      |

|           |          |           |
|-----------|----------|-----------|
| LL        | SSSSSSSS | TTTTTTTTT |
| LL        | SSSSSSSS | TTTTTTTTT |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SSSSSS   | TT        |
| LL        | SSSSSS   | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LLLLLLLLL | SSSSSSSS | TT        |
| LLLLLLLLL | SSSSSSSS | TT        |

```

57 .SBTTL MACRO CALLS AND LOCAL DATA
58 .SBTTL MACRO DEFINITIONS
59
60 ; LOCAL MACROS
61 ;
62
63 .MACRO PRINT TEXT
64 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
65 CALL $PRINT ; PRINT MESSAGE
66 .ENDM PRINT
67
68 .MACRO ERRORS$ TEXT
69
70 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
71 ; MESSAGE STRING.
72 ;
73 .IF DIF <TEXT><R0>
74 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
75 .ENDC
76 CALL $CFERR ; PRINT ERROR MESSAGE
77 .ENDM ERRORS$
78
79 .MACRO EPRINT TEXT
80
81 ; PRINT TEXT ON ERROR LUN
82 ;
83 .IF DIF <TEXT><R0>
84 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
85 .ENDC
86 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
87 .ENDM EPRINT
88
89 ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
90 ;
91 .MACRO $$$PTN ROU
92 .MACRO 'ROU OFS,OCTL,OPT,?L
93 .IF NB <OPT>
94 MOV OPT,R2
95 CALL $PTOPT
96 BCS L
97 .IFTF
98 MOV OFS,R2
99 .IF NB <OCTL>
100 MOV SP,R1
101 .IFF
102 CLR R1
103 .ENDC ; IF NB <OCTL>
104 CALL 'ROU
105 .IFT ; IF NB <OPT>
106
107 L:
108 .ENDC ; IF NB <OPT>
109 .ENDM ; 'ROU
110 .ENDM ; $$$PTN
111
112 $$$PTN $PTNUM
113 $$$PTN $PTNMB

```

```

111 .SBTTL STATE TABLES
112
113 ;****
114 ; INITIALIZE TABLE GENERATION
115 ;****
116
117
118 000202 ISTAT$ EVTSTB,EVTKTB
119
120
121 000202 STATES$
122 000202 TRANS$ $NUMBR,,SETCLS
123
124 000202 STATES$
125 000202 TRANS$ '<','>','$MOVPT
126
127 000202 STATES$
128 000202 TRANS$ LA,, $MOVPT
129
130 000202 STATES$
131 000202 TRANS$!EVTMSK
132
133 000202 STATES$
134 000202 TRANS$ '<','>','$MOVPT
135
136 000202 STATES$
137 000202 TRANS$ $STRNG,EXIT,SETSNK
138
139 ;
140 ; EXIT STATE
141 ;
142 000202 STATES$ EXIT
143 000202 TRANS$ $EOS,$EXIT
144
145 ;
146 ; EVENT MASK SUB-EXPRESSION
147 ;
148 000202 STATES$ EVTMSK
149 000202 TRANS$ $NUMBR,,SETMSK
150
151 000202 STATES$
152 000202 TRANS$ RA,$EXIT,$MOVPT
153 000202 TRANS$ '<','>','$EVTMSK,$MOVPT
154
155 000202 STATES$

```

```

46 .SBTTL MACRO CALLS AND LOCAL DATA
47
48 .MCALL SAVRG,RESRG
49
50 ;
51 ; LOCAL DATA
52 ;
53 ;.NLIST BEX
54 000000 011 106 111 FILDF: .ASCII <11>/FIL$DF/<11>
55 000010 .FILLN=-FILDF
56 .EVEN
57 .LIST BEX
58
59 ;
60 ; Define FIL$DF template
61 ;
62 000000 .ASECT
63 000000 . = 0
64 000000 .BLKW 1 ; Link
65 000002 .BLKW 1 ; Flags
66 000004 F$SEC: .BLKW 1 ; Secondary loader file name
67 000006 F$TER: .BLKW 1 ; Tertiary loader file name
68 000010 F$LD: .BLKW 1 ; Load file name
69 000012 F$DIA: .BLKW 1 ; Diagnostics file name
70 000014 F$DUM: .BLKW 1 ; Dump file name
71 000016 F$LEN=.
72 000010 .PSECT
73
74 ;
75 ; Define file descriptor offsets
76 ;
77
78 000036 MXFSP = 30. ; Undivided Filespec length
79
80 000016 .ASECT
81 000000 . = 0
82 000000 F.LNK: .BLKW 1 ; Link word
83 000002 F.CNT: .BLKW 1 ; Use count
84 000004 .BLKB 1 ; Filespec string length
85 000005 F.DEV: .BLKB MXFSP ; Filespec string (ASCII)
86 000043 F.LEN = .
87 000010 .PSECT
88

```

UPDFEA - CFE UPDATE FEATURES DE MACRO V05.03b Saturday 29-Jun-85 05:52 Page 7-1

## Symbol table

```

CS.FEA= ***** GX FEAKTB 000000RG 003 $CLNUP= ***** GX $LAMDA= 000000 $SUBXP= 000010
CS.MOD= ***** GX FEASTB 000000RG 002 $CNVO = ***** GX $MOVPT= ***** GX $TMLST= ***** GX
C.DEV = 000004 F.NRBD= ***** GX $DIGIT= 000024 $MOVST= ***** GX $UPFEA 000002RG
C.LEN = 000014 SETWD3 000114R $DNUMB= 000014 $NUMBR= 000002 $$$FLG= 177777
C.STS = ***** GX TEMP 000000R $EOS = 000012 $OBUF = ***** GX $$$KEY= 177777
C.WD1 000006 $ALPHA= 000022 $ERROR= ***** GX $OFDB = ***** GX $$$STA= 000000
C.WD2 000010 $ANY = 000020 $EXIT = 000000 $RAD50= 000016 .CONER= ***** GX
C.WD3 000012 $BLANK= 000006 $FAIL = 177777 $STRNG= 000004 .TPARS= ***** GX
EXIT 000034R 002 $BUFI = ***** GX
. ABS. 000014 000 (RW,I,GBL,ABS,OVR)
 000142 001 (RW,I,LCL,REL,CON)
$STATE 000042 002 (RW,D,LCL,REL,CON)
$KTAB 000000 003 (RW,D,LCL,REL,CON)
$KSTR 000000 004 (RW,D,LCL,REL,CON)
Errors detected: 0

```

## \*\*\* Assembler statistics

```

Work file reads: 0
Work file writes: 0
Size of work file: 10406 Words (41 Pages)
Size of core pool: 14440 Words (55 Pages)
Operating system: RSX-11M/PLUS

```

Elapsed time: 00:00:15.07

SY:UPDFEA.V2,[132,134]UPDFEA/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDFEA

```

308 .SBTTL SETPRI - SET THE PRIORITY
309 :+
310 : *** - SETPRI - SET THE PROCESS PRIORITY
311 :
312 : INPUT:
313 : .PNUMB = PRIORITY
314 : .PNUMH = PRIORITY (HIGH ORDER)
315 :
316 : OUTPUT:
317 :
318 : -
319
320 SETPRI:
321 000500 016702 177274 MOV TEMP,R2 ; GET START OF TEMPLATE
322 000504 005001 CLR R1 ; GET PRIORITY WITHOUT SIGN EXTEND
323 000506 156201 000010 BISB C,PRI(R2),R1 ;
324 000512 016700 000000G MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
325 000516 004767 000000G CALL $CNVD ; CONVERT THE NUMBER TO OCTAL
326 000522 010067 000000G MOV R0,$OBUF ; UPDATE THE POINTER INTO THE OUTPUT BUFFER
327 000526 000207 RETURN

```

```

LOCAL DATA
92
93
94
95
96
97
98
99
100
101 000000
102 000000
103 000000
104 000002
105 000004
106 000005
107 000000
108
109 000000

.SBTTL LOCAL DATA
;****
; LOCAL DATA
;****

; DEFINE THE LOGGING STATE BUFFER BLOCK
;
.ASECT
.=0
.BLKW 1 ; C.LNK
.BLKW 1 ; C.STS
C.STA: .BLKB 1 ; LOGGING STATE
C.LEN=. ; LENGTH OF THE BLOCK
.PSECT

NEXT: .BLKW 1 ; ADDRESS OF NEXT LOCATION IN TEMPLATE

```



```

159 .SBTTL $UPNOD - UPDATE NODE DEFINITION
160 +
161 *** - $UPNOD - UPDATE NODE DEFINITION
162
163 INPUT:
164 R3-R5 - TPARS REGISTERS
165
166 OUTPUT:
167 C-BIT = SUCCESS/FAILURE
168
169 -
170
171 $UPNOD::
172 000002 012700 000000G MOV #STMLST,R0 ; POINT AT START OF TEMPLATES
173 000006 010001 MOV R0,R1 ; SAVE CURRENT TEMPLATE
174 000010 011000 MOV (R0),R0 ; GET NEXT TEMPLATE
175 000012 001436 BEQ 101$; IF EQ, END OF TEMPLATES
176 000014 122760 000000G CMPB #CS.NOD,C.STS(R0) ; IS THIS A NODE TEMPLATE ?
177 000022 001371 BNE 10$; IF NE, NO .. KEEP LOOKING
178
179 ;
180 ; ASSUME THIS IS THE RIGHT TEMPLATE
181 ;
182 000024 011011 MOV (R0),(R1) ; REMOVE TEMPLATE FROM CHAIN
183 000026 010067 177746 MOV R0,TEMP ; SAVE TEMPLATE ADDRESS
184 000032 032760 000000G 000000G BIT #CS.MOD,C.STS(R0) ; WAS THIS TEMPLATE MODIFIED ?
185 000040 001004 BNE 15$; IF NE, YES .. UPDATE THE DEFINITION
186 000042 012767 000000G 000000C MOV #BUFI,$OFDB+F.NRBD+2 ; ELSE, OUTPUT THE INPUT BUFFER
187 000050 000416 BR 30$; AND LEAVE
188 000052 004767 000000G 15$: CALL $MOVST ; MOVE FIRST FIELD INTO OUTPUT BUFFER
189 000056 010546 MOV R5,-(SP) ; SAVE R5
190 000062 012702 000000' CLR R1 ; IGNORE BLANKS
191 000066 012705 000000' MOV #NODKTB,R2 ; KEYWORD TABLE
192 000072 004767 000000G MOV #NODSTB,R5 ; STATE TABLE
193 000076 103002 CALL .TPARS ; PARSE THE REST OF THE LINE
194 000100 005267 000000G BCC 20$; NORMAL RETURN IF NO ERROR
195 000104 012605 INC $ERROR ; INDICATE SYNTAX ERROR
196 000106 000207 MOV (SP)+,R5 ; RESTORE R5
197
198 ;
199 ; ERROR CONDITIONS
200 000110 000167 000000G 101$: JMP .CONER ; CONSISTENCY ERROR

```

```

141 .SBTTL $UPOBJ - SETUP OBJECT DEFINITION
142
143 *** - $UPOBJ - SETUP OBJECT DEFINITION
144
145 INPUT:
146 R3-R5 - TPARS REGISTERS
147
148 OUTPUT:
149 C-BIT = SUCCESS/FAILURE
150
151 -
152
153 $UPOBJ::
154 000150 MOV #BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
155 000150 012767 000027' 000000C MOV R5,-($P) ; SAVE CURRENT TPARS STATE
156 000156 010546 5$: MOV #TMLST,RO ; POINT AT START OF TEMPLATES
157 000160 012700 000000G 10$: MOV RO,R1 ; SAVE ADDRESS OF CURRENT TEMPLATE
158 000164 010001 MOV (RO),RO ; GET ADDRESS OF NEXT TEMPLATE
159 000166 011000 BEQ 30$; IF EQ, END OF TEMPLATES
160 000170 001451 CMPB #CS.OBJ,C.STS(RO) ; IS THIS AN OBJECT DEFINITION
161 000172 122760 BNE 10$; IF NE, NO .. KEEP LOOKING
162 000200 001371
163
164 : FOUND A TEMPLATE
165
166 MOV (RO),(R1) ; REMOVE TEMPLATE FROM CHAIN
167 000202 011011 MOV RO,TEMP ; SAVE TEMPLATE ADDRESS
168 000204 010067 177570 MOV #BUFF,RO ; POINT AT START OF OUTPUT BUFFER
169 000210 012700 000027' MOV #OBJDF,R1 ; POINT AT START OF OBJECT DEFINITION
170 000214 012701 000017' MOV #OBJLN,R2 ; GET LENGTH OF OBJECT DEFINITION
171 000220 012702 000010 CALL $MVASC ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
172 000224 004767 000000G MOV RO,$OBJUF ; SAVE CURRENT POSITION IN OUTPUT BUFFER
173 000230 010067 000000G CLR R1 ; IGNORE BLANKS
174 000234 005001 MOV #OBJKT,R2 ; KEYWORD TABLE
175 000236 012702 000000' MOV #OBJLEN,R3 ; LENGTH OF OBJECT DEFINITION
176 000242 012703 000014 MOV #OBJDEF,R4 ; ADDRESS OF OBJECT DEFINITION
177 000246 012704 000002' MOV #OBJSTB,R5 ; STATE TABLE
178 000252 012705 000000' CALL TPARS ; PARSE THE REST OF THE LINE
179 000256 004767 000000G BCC 20$; NORMAL RETURN IF NO ERROR
180 000262 103003 INC $ERROR ; INDICATE SYNTAX ERROR
181 000264 005267 000000G BR 30$; AND LEAVE
182 000270 000411 20$: MOV $OBJUF,R1 ; CALCULATE LENGTH OF BUFFER
183 000272 016701 000000G SUB #BUFF,R1
184 000276 162701 MOV R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
185 000302 010167 000000C CALL $WDATA ; WRITE THE RECORD TO THE OUTPUT FILE
186 000306 004767 000000G BR 30$; AND LOOK FOR MORE OBJECT TEMPLATES
187 000312 000722 30$: MOV (SP)+,R5 ; RESTORE R5
188 000314 012605 MOV #BUFI,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE "END$DF" LINE
189 000316 012767 RETURN
190 000324 000207

```

```

53
54
55 ;*****
56 ; MACRO CALLS
57 ;*****
58 .MCALL ISTAT$,STAT$,TRAN$
59
60 ;*****
61 ; LOCAL DATA
62 ;*****
63
64 000000 TEMP: .BLKW 1 ; ADDRESS OF DLC TEMPLATE
65 .NLIST BEX
66 000002 124 117 120 TOPSW: .ASCII /TOP/ ; TOP DOWN LOADING SWITCH
67 000003 TOPPL=-TOPSW
68 000005 123 102 120 SBPSW: .ASCII /SBPOOL/ ; Server Base pool switch
69 000006 SBPL=-SBPSW
70 .EVEN
71
72 ;
73 ; DEFINE PARTITION TEMPLATE
74 ;
75 000000 .ASECT
76 000000 .=0
77 000002 .BLKW 1 ; C.LNK
78 000004 .BLKW 1 ; C.STS
79 000010 C.NAM0: .BLKW 2 ; POOL PARTITION NAME (RAD50)
80 000014 C.NAM1: .BLKW 2 ; DYNAMIC PARTITION TO CREATE POOL FROM (RAD50)
81 000016 C.EXT: .BLKW 1 ; POOL EXTENSION SIZE
82 C.ALL: .BLKW 1 ; NUMBER OF BLOCKS TO ALLOCATE FOR SINGLE WORD POOL
83 000014 C.LEN=.
84 .PSECT

```

UPDPSN - UPDATE PSN IN CETAB    MACRO V05.03b Saturday 29-Jun-85 05:55 <sup>I 16</sup>  
Table of contents

4- 57    MACRO DEFINITIONS  
5- 122    LOCAL DATA  
6- 161    SUPPSN - UPDATE PSN DEFINITION

310  
 311  
 312  
 313  
 314  
 315  
 316  
 317  
 318  
 319  
 320  
 321  
 322  
 323  
 324  
 325  
 326  
 327  
 328  
 329  
 330

000674  
 000674 016702 177100  
 000700 005001  
 000702 156201 000012  
 000706 016700 000000G  
 000712 004767 000000G  
 000716 010067 000000G  
 000722 000207

```
.SBTTL SETNLT - STORE NUMBER OF LINE TABLES
*** - SETNLT - STORE NUMBER OF LINE TABLES
INPUT:
.PNUMB = NUMBER OF LINE TABLE TO BE SET IN PROCESS SPACE
TEMP = TEMPLATE ADDRESS
OUTPUT:
C.NLT IN TEMPLATE MOVED INTO OUTPUT BUFFER
--
SETNLT:
MOV TEMP,R2 ; GET THE TEMPLATE ADDRESS
CLR R1 ; GET THE LINE TABLE COUNT
BTSB C.NLT(R2),R1 ; ..WITHOUT SIGN EXTEND
MOV $0BUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
CALL $CNVD ; CONVERT NUMBER TO DECIMAL
MOV R0,$0BUF ; UPDATE POINTER INTO OUTPUT BUFFER
RETURN
```

```

182 .SBTTL SETLKS - NUMBER OF LOGICAL LINKS PROCESS SHOULD SUPPORT
183
184 ;+
185 *** - SETLKS - SETUP NUMBER OF LOGICAL LINKS
186 ;
187 INPUT:
188 TEMP = TEMPLATE ADDRESS
189 $OBUF = CURRENT POSITION IN OUTPUT BUFFER
190 ;
191 OUTPUT:
192 ;
193 -
194 SETLKS:
195 000114 MOV TEMP,R2 ; GET START OF TEMPLATE
196 000114 016702 177660 MOV C.LKS(R2),R1 ; GET NUMBER OF LOGICAL LINKS
197 000120 016201 000010 MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
198 000124 016700 000000G CALL $CNVD ; CONVERT NUMBER TO DECIMAL
199 000130 004767 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
200 000140 000207 000000G RETURN

```

## SYMBOL CROSS REFERENCE

CREF 04.00

| SYMBOL    | VALUE       | REFERENCES |
|-----------|-------------|------------|
| \$RAD50   | = 000016    | #5-67      |
| \$RONLY   | = *****     | 5-67 5-67  |
| \$STRNG   | = 000004    | #5-67      |
| \$SUBXP   | = 000010    | #5-67      |
| \$TMLST   | = ***** GX  | 6-152      |
| \$UPDEC   | = 000002 RG | #6-151     |
| \$\$\$FLG | = 177777    | #5-67      |
| \$\$\$KEY | = 177777    | #5-67      |
| .CONER    | = ***** GX  | 6-180      |
| .TPARS    | = ***** GX  | 6-172      |

```

314 .SBTTL SETNCT - STORE NUMBER OF CONTROLLER TABLES
315 :+
316 :*** - SETNCT - STORE NUMBER OF CONTROLLER TABLES
317 :
318 :INPUT:
319 : TEMP = TEMPLATE ADDRESS
320 :
321 :OUTPUT:
322 : C.NCT IN TEMPLATE MOVED INTO OUTPUT BUFFER
323 :
324 :-
325
326 SETNCT:
327 MOV TEMP,R2 ; GET THE TEMPLATE ADDRESS
328 CLR R1 ; GET THE CONTROLLER TABLE COUNT
329 BISB C.NCT(R2),R1 ; ..WITHOUT SIGN EXTEND
330 MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
331 CALL $CNVD ; CONVERT NUMBER TO DECIMAL
332 MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
333 RETURN
334
335
336 000001 .END

```



UPDDSA      CREATED BY    MACRO    ON 29-JUN-85 AT 05:50      PAGE 1      I 5  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE       | REFERENCES                             |
|---------|-------------|----------------------------------------|
| CS.NUM  | = ***** GX  | 6-176                                  |
| CS.SHI  | = ***** GX  | 6-178                                  |
| CS.SLO  | = ***** GX  | 6-177                                  |
| C.CUG   | = 000022    | #5-143      6-179                      |
| C.LEN   | = 000026    | #5-144                                 |
| C.NUM   | = 000004    | #5-140      6-176                      |
| C.SHI   | = 000020    | #5-142      6-178                      |
| C.SLO   | = 000016    | #5-141      6-177                      |
| DA\$CUG | = ***** GX  | 6-179                                  |
| DSADF   | = 000000 R  | #5-147      5-148      6-172           |
| DSALN   | = 000010    | #5-148      6-173                      |
| F.NRBD  | = ***** GX  | *6-166      *6-182      *6-184         |
| \$BUFI  | = ***** GX  | 6-184                                  |
| \$MVASC | = ***** GX  | 6-174                                  |
| \$OBUF  | = ***** GX  | *6-175      6-180                      |
| \$OFDB  | = ***** GX  | *6-166      *6-182      *6-184         |
| \$PTNUM | = ***** GX  | 6-177      6-178                       |
| \$PTOPT | = ***** GX  | 6-176      6-177      6-178      6-179 |
| \$PTR50 | = ***** GX  | 6-176                                  |
| \$UPDSA | = 000010 RG | #6-165                                 |
| \$WDATA | = ***** GX  | 6-183                                  |
| .BUFF   | = ***** GX  | 6-166      6-171      6-181            |
| .COMMA  | = ***** GX  | *6-170                                 |
| .FLAGS  | = ***** GX  | *6-169                                 |
| .TEMP   | = ***** GX  | *6-167                                 |

UPDDSA      CREATED BY    MACRO    ON 29-JUN-85 AT 05:50      PAGE 2      J 5  
 MACRO CROSS REFERENCE      CREF    04.00

UPDDST - UPDATE DESTINATION DES MACRO V05.03b Saturday 29-Jun-85 05:51 I 6  
Table of contents

|    |     |                                         |
|----|-----|-----------------------------------------|
| 4- | 57  | MACRO CALLS AND LOCAL DATA              |
| 4- | 58  | MACRO DEFINITIONS                       |
| 5- | 131 | LOCAL DATA                              |
| 6- | 154 | \$UPDST - UPDATE DESTINATION DESCRIPTOR |

```

114 ; PUT RAD50 TEMPLATE FIELD TO BUFFER
115 ;
116 .MACRO $PTR50 OFS,WC,OPT,?L
117 .IF NB <OPT>
118 MOV OPT,R2
119 CALL $PTOPT
120 BCS L
121 .IFTF
122 MOV OFS,R2
123 MOV WC,R1
124 CALL $PTR50
125 .IFT
126 L:
127 .ENDC
128 .ENDM
129 ;
130 ; MACRO CALLS
131 ;
132 .MCALL CALLR
133

```

```

157 .SBTTL $UPEVT - SETUP EVENT DEFINITION
158 ;+
159 *** - $UPEVT - SETUP EVENT DEFINITION
160
161 INPUT:
162 R3-R5 - TPARS REGISTERS
163
164 OUTPUT:
165 C-BIT = SUCCESS/FAILURE
166
167 -
168
169 000202 $UPEVT::
170 000202 012767 000062' 000000C MOV #BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
171 000210 010546 MOV R5,-(SP) ; SAVE CURRENT TPARS STATE
172 000212 012700 000000G 5$: MOV #TMLST,R0 ; POINT AT START OF TEMPLATES
173 000216 010001 10$: MOV R0,R1 ; SAVE ADDRESS OF CURRENT TEMPLATE
174 000220 011000 MOV (R0),R0 ; GET ADDRESS OF NEXT TEMPLATE
175 000222 001451 BEQ 30$; IF EQ, END OF TEMPLATES
176 000224 122760 000000G 000000G CMPB #CS.EVT,C.STS(R0) ; IS THIS AN EVENT DEFINITION
177 000232 001371 BNE 10$; IF NE, NO .. KEEP LOOKING
178
179 ; FOUND A TEMPLATE
180
181 000234 011011 MOV (R0),(R1) ; REMOVE TEMPLATE FROM CHAIN
182 000236 010067 177536 MOV R0,TEMP ; SAVE TEMPLATE ADDRESS
183 000242 012700 000062' MOV #BUFF,R0 ; POINT AT START OF OUTPUT BUFFER
184 000246 012701 000030' MOV #EVTDF,R1 ; POINT AT START OF EVENT DEFINITION
185 000252 012702 000010 MOV #EVTLN,R2 ; GET LENGTH OF EVENT DEFINITION
186 000256 004767 000000G CALL $MVASC ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
187 000262 010067 000000G MOV R0,$OBUF ; SAVE CURRENT POSITION IN OUTPUT BUFFER
188 000266 005001 CLR R1 ; IGNORE BLANKS
189 000270 012702 000000' MOV #EVTKTBL,R2 ; KEYWORD TABLE
190 000274 012703 000023' MOV #EVTLEN,R3 ; LENGTH OF EVENT DEFINITION
191 000300 012704 000004' MOV #EVTDEF,R4 ; ADDRESS OF EVENT DEFINITION
192 000304 012705 000000' MOV #EVTSTBL,R5 ; STATE TABLE
193 000310 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
194 000314 103003 BCC 20$; NORMAL RETURN IF NO ERROR
195 000316 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
196 000322 000411 BR 30$; AND LEAVE
197 000324 016701 000000G 20$: MOV $OBUF,R1 ; CALCULATE LENGTH OF BUFFER
198 000330 162701 000062' SUB #BUFF,R1
199 000334 010167 000000C MOV R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
200 000340 004767 000000G CALL $WDATA ; WRITE THE RECORD TO THE OUTPUT FILE
201 000344 000722 BR 30$; AND LOOK FOR MORE EVENT TEMPLATES
202 000346 012605 MOV (SP)+,R5 ; RESTORE R5
203 000350 012767 000000G 000000C MOV #BUFF1,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE 'END$DF' LINE
204 000356 000207 RETURN

```

```

100
101
102 .SBTTL $UPFIL - SETUP REMOTE DEFINITION
103
104 *** - $UPFIL - SETUP REMOTE DEFINITION
105
106 INPUT:
107 R5 = FIL$DF TEMPLATE ADDRESS
108
109 OUTPUT:
110 C-BIT = SUCCESS/FAILURE
111
112
113
114 $UPFIL::
115 JSR R5,$SAVRG ; SAVE REGISTERS R3-R5
116 MOV $OFDB+F.NRBD+2,-(SP) ; SAVE REM$DF ADDRESS
117 MOV #,BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
118 MOV R5,TEMP ; SAVE TEMPLATE ADDRESS
119 TST (R5)+ ; POINT TO C.STS
120 MOV R5,.FLAGS ; SAVE ADDRESS OF FLAGS WORD
121 MOV #,BUFF,R0 ; POINT TO BUFFER
122 MOV #FILD,R1 ; POINT AT START OF FIL DEFINITION
123 MOV #FILLN,R2 ; GET LENGTH OF FIL DEFINITION
124 CALL $MVASC ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
125 MOV R0,$OBUF ; SAVE CURRENT POSITION IN OUTPUT BUFFER
126
127 JSR R5,PUTFIL ; SECONDARY BOOT FILE
128 .WORD F$SEC ; ...
129
130 JSR R5,PUTFIL ; TERTIARY BOOT FILE
131 .WORD F$TER ; ...
132
133 JSR R5,PUTFIL ; LOAD FILE
134 .WORD F$LD ; ...
135
136 JSR R5,PUTFIL ; DIAGNOSTIC FILE
137 .WORD F$DIA ; ...
138
139 JSR R5,PUTFIL ; DUMP FILE
140 .WORD F$DUM ; ...
141
142 MOV $OBUF,R1 ; CALCULATE LENGTH OF BUFFER
143 CMPB -(R1),#<','> ; TRAILING COMMA ?
144 BEQ 10$; IF EQ, YES
145 INC R1 ; ELSE BACKED UP TOO FAR
146
147 SUB #,BUFF,R1 ;
148 MOV R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
149 CALL $WDATA ; WRITE THE RECORD TO THE OUTPUT FILE
150 MOV (SP)+,$OFDB+F.NRBD+2 ; RESTORE REM$DF ADDRESS
151 RETURN

```

UPDFEA      CREATED BY    MACRO    ON 29-JUN-85 AT 05:52      PAGE 1      I 10  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE       | REFERENCES                   |
|-----------|-------------|------------------------------|
| CS.FEA    | = ***** GX  | 6-143                        |
| CS.MOD    | = ***** GX  | 6-150                        |
| C.DEV     | = 000004    | #4-82                        |
| C.LEN     | = 000014    | #4-86                        |
| C.STS     | = ***** GX  | 6-143      6-150             |
| C.WD1     | = 000006    | #4-83                        |
| C.WD2     | = 000010    | #4-84                        |
| C.WD3     | = 000012    | #4-85      7-187             |
| FEAKTB    | = 000000 RG | #5-95      6-157             |
| FEASTB    | = 000000 RG | #5-95      6-158             |
| F.NRBD    | = ***** GX  | *6-152                       |
| SETWD3    | = 000114 R  | #7-184                       |
| TEMP      | = 000000 R  | #4-74      *6-149      7-186 |
| \$ALPHA   | = 000022    | #5-95                        |
| \$ANY     | = 000020    | #5-95                        |
| \$BLANK   | = 000006    | #5-95                        |
| \$BUFI    | = ***** GX  | 6-152                        |
| \$CNVO    | = ***** GX  | 7-188                        |
| \$DIGIT   | = 000024    | #5-95                        |
| \$DNUMB   | = 000014    | #5-95                        |
| \$EOS     | = 000012    | #5-95                        |
| \$ERROR   | = ***** GX  | *6-161                       |
| \$EXIT    | = 000000    | #5-95                        |
| \$FAIL    | = 177777    | #5-95                        |
| \$GPRM    | = *****     | 5-95                         |
| \$LAMDA   | = 000000    | #5-95                        |
| \$MOVST   | = ***** GX  | 6-154                        |
| \$NUMBR   | = 000002    | #5-95                        |
| \$OBUF    | = ***** GX  | 7-185      *7-189            |
| \$OFDB    | = ***** GX  | *6-152                       |
| \$RAD50   | = 000016    | #5-95                        |
| \$RONLY   | = *****     | 5-95      5-95               |
| \$STRNG   | = 000004    | #5-95                        |
| \$SUBXP   | = 000010    | #5-95                        |
| \$TMLST   | = ***** GX  | 6-139                        |
| \$UPFEA   | = 000002 RG | #6-138                       |
| \$\$\$FLG | = 177777    | #5-95                        |
| \$\$\$KEY | = 177777    | #5-95                        |
| .CONER    | = ***** GX  | 6-167                        |
| .TPARS    | = ***** GX  | 6-159                        |

```

329
330
331
332
333
334
335
336
337
338
339
340
341 000530
342 000530 016702 177244
343 000534 005001
344 000536 156201 000011
345 000542 016700 000000G
346 000546 004767 000000G
347 000552 010067 000000G
348 000556 000207

 .SBTTL SETLIN - SET LINE NUMBER
 :+
 *** - SETLIN - SETUP LINE NUMBER
 :
 INPUT:
 :.PNUMB = LINE NUMBER
 :.PNUMH = LINE NUMBER (HIGH ORDER)
 :
 OUTPUT:
 :-
 SETLIN:
 MOV TEMP,R2 ; GET START OF TEMPLATE
 CLR R1 ; GET NUMBER OF LINES WITHOUT SIGN EXTEND
 BISB C.LIN(R2),R1 ;
 MOV $0BUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
 CALL $CNVD ; CONVERT BINARY TO OCTAL
 MOV R0,$0BUF ; UPDATE THE BUFFER POINTER
 RETURN

```

```
111 .SBTTL STATE TABLES
112
113 ; *****
114 ; INITIALIZE TABLES
115 ; *****
116 000002 ISTAT$ LOGSTB,LOGKTB
117
118 ;
119 ; PROCESS 'LOG$ST'
120 ;
121 000002 STATE$
122 000002 TRANS$ $NUMBR,,SETNUM
123
124 000002 STATE$
125 000002 TRANS$ $EOS,$EXIT,$CLNUP
126
127 000002 STATE$
```



```

202 .SBTTL SETNAM - SET UP NODE NAME
203 +
204 *** - SETNAM - SETUP NODE NAME
205 :
206 : INPUT:
207 : .PSTPT = START ADDRESS OF THE NODE NAME IN ASCII
208 : .PSTCN = NUMBER OF CHARACTERS IN THE NODE NAME
209 : TEMP = TEMPLATE ADDRESS
210 : TEMP+C.NAM = NODE NAME
211 :
212 : OUTPUT:
213 :
214 : -
215
216 000114 SETNAM: MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
217 000114 016700 000000G MOV TEMP,R2 ; GET TEMPLATE ADDRESS
218 000120 016702 177654 MOV C.NAM(R2),R1 ; GET RAD50 NODE NAME
219 000124 016201 000004 CALL $CSTA ; CONVERT RAD50 TO ASCII
220 000130 004767 000000G MOV TEMP,R2 ; GET TEMPLATE ADDRESS
221 000134 016702 177640 MOV C.NAM+2(R2),R1 ; GET 2ND HALF OF NODE NAME
222 000140 016201 000006 CALL $CSTA ; CONVERT RAD50 TO ASCII
223 000144 004767 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
224 000150 010067 000000G
225 000154 000207

```

190  
 191  
 192  
 193  
 194  
 195  
 196  
 197  
 198  
 199  
 200  
 201  
 202  
 203  
 204  
 205  
 206  
 207  
 208  
 209  
 210  
 211  
 212

000326  
 000326 016700 000000G  
 000332 016702 177442  
 000336 005001  
 000340 156201 000004  
 000344 004767 000000G  
 000350 010067 000000G  
 000354 000207

```

 .SBTTL SETYPE - WRITE OUT OBJECT TYPE
 *** - SETYPE - WRITE OUT OBJECT TYPE
 INPUT:
 .PNUMB = THE OBJECT NUMBER
 .PNUMH = THE OBJECT NUMBER (HIGH ORDER)
 $OBUF = CURRENT POSITION IN OUTPUT LINE
 TEMP = TEMPLATE ADDRESS
 OUTPUT:
 C-BIT = SUCCESS/FAILURE
 -
 SETYPE:
 MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN LINE
 MOV TEMP,R2 ; POINT AT TEMPLATE
 CLR R1 ; GET OBJECT TYPE WITHOUT SIGN EXTEND
 BISB C.TYP(R2),R1 ;
 CALL $CNVD ; CONVERT BINARY TO DECIMAL
 MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT LINE
 RETURN

```

```

86
87 ;*****
88 ; INITIALIZE TABLE GENERATION
89 ;*****
90 000014 ISTAT$ PARSTB,PARKTB
91
92 000014 STATES$
93 000014 TRANS$ $RAD50,NEXT,SETNMO ; pool name
94 000014 TRANS$ $LAMDA,,SETNMO
95
96 000014 STATES$ NEXT
97 000014 TRANS$ <'>,, $MOVPT
98
99 000014 STATES$
100 000014 TRANS$ $RAD50,,SETNM1 ; partition
101
102 000014 STATES$
103 000014 TRANS$ <'>,, $MOVPT
104
105 000014 STATES$
106 000014 TRANS$ $NUMBR,, $MOVPT ; block extension
107
108 000014 STATES$
109 000014 TRANS$ <'>,TOP, $MOVPT
110 000014 TRANS$ $LAMDA,EXIT
111
112 000014 STATES$ TOP
113 000014 TRANS$ 'TOP',UMRS,SETTOP ; loading info
114 000014 TRANS$ $LAMDA,,SETTOP
115
116 000014 STATES$ UMRS
117 000014 TRANS$ $EOS,EXIT
118 000014 TRANS$ <'>,, $MOVPT
119
120 000014 STATES$
121 000014 TRANS$ $NUMBR,ALLC, $MOVPT ; UMR unmapped blocks
122 000014 TRANS$ 'NONE',ALLC, $MOVPT
123 000014 TRANS$ $LAMDA
124
125
126 000014 STATES$ ALLC
127 000014 TRANS$ $EOS,EXIT
128 000014 TRANS$ <'>,, $MOVPT
129
130 000014 STATES$
131 000014 TRANS$ $NUMBR,,SETALL ; pool byte extension
132
133 000014 STATES$ SBCHK
134 000014 TRANS$ $EOS,EXIT
135 000014 TRANS$ <'>,, $MOVPT
136
137 000014 STATES$
138 000014 TRANS$ 'SBPOOL',EXIT,SETSBP
139
140 ; EXIT STATE
141 ;
142 000014 STATE$ EXIT

```

.TITLE UPDPSN - UPDATE PSN IN CETAB  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO UPDATE THE PSN DEFINITION

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0

```

332 .SBTTL SETNCT - STORE NUMBER OF CONTROLLER TABLES
333 :+
334 *** - SETNCT - STORE NUMBER OF CONTROLLER TABLES
335 :
336 INPUT:
337 TEMP = TEMPLATE ADDRESS
338 :
339 OUTPUT:
340 C.NCT IN TEMPLATE MOVED INTO OUTPUT BUFFER
341 :
342 :-
343
344 000724 SETNCT:
345 000724 016702 177050 MOV TEMP,R2 ; GET THE TEMPLATE ADDRESS
346 000730 005001 CLR R1 ; GET THE CONTROLLER TABLE COUNT
347 000732 156201 000013 BLSB C.NCT(R2),R1 ; ..WITHOUT SIGN EXTEND
348 000736 016700 000000G MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
349 000742 004767 000000G CALL $CNVD ; CONVERT NUMBER TO DECIMAL
350 000746 010067 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
351 000752 000207 RETURN
352
353
354 000001 .END

```

```

202 .SBTTL SETNDC - NUMBER OF NODE COUNTERS
203 ;+
204 ; *** - SETNDC - SETUP NUMBER OF NODE COUNTERS
205 ;
206 ; INPUT:
207 ; TEMP = TEMPLATE ADDRESS
208 ; $OBUF = CURRENT POSITION IN OUTPUT BUFFER
209 ;
210 ; OUTPUT:
211 ;
212 ; -
213
214 000142 SETNDC:
215 000142 MOV TEMP,R2 ; GET START OF TEMPLATE
216 000146 016702 177632 MOV C.NDC(R2),R1 ; GET NUMBER OF LOGICAL LINKS
217 000152 016201 000006 MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
218 000156 004767 000000G CALL $CNVD ; CONVERT NUMBER TO DECIMAL
219 000162 010067 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
220 000166 000207

```

UPDDEC      CREATED BY    MACRO    ON 29-JUN-85 AT 05:49      PAGE 3      J 3  
 MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$ | #5-67  |        |        |        |        |        |        |        |        |        |
| DECTP\$ | #4-50  | 4-52   |        |        |        |        |        |        |        |        |
| ISTAT\$ | #4-50  | 5-67   |        |        |        |        |        |        |        |        |
| MSGDF\$ | #4-50  | 4-53   |        |        |        |        |        |        |        |        |
| MTRAN\$ | #5-67  |        |        |        |        |        |        |        |        |        |
| STAT\$  | #4-50  | 5-69   | #5-72  | #5-75  | #5-78  | #5-81  | #5-84  | #5-87  | #5-90  | #5-93  |
|         | #5-96  | #5-99  | #5-102 | #5-105 | #5-108 | #5-111 | #5-114 | #5-117 | #5-120 | #5-123 |
|         | #5-126 | #5-129 | #5-134 | #5-137 |        |        |        |        |        |        |
| TRAN\$  | #4-50  | #5-70  | #5-73  | #5-76  | #5-79  | #5-82  | #5-85  | #5-88  | #5-91  | #5-94  |
|         | #5-97  | #5-100 | #5-103 | #5-106 | #5-109 | #5-112 | #5-115 | #5-118 | #5-121 | #5-124 |
|         | #5-127 | #5-130 | #5-135 |        |        |        |        |        |        |        |

|                  |                       |                       |                   |                    |
|------------------|-----------------------|-----------------------|-------------------|--------------------|
| BITS 000016R     | 002 NCT 000104R       | 002 ZF.TIM= ***** GX  | \$ERROR= ***** GX | \$UPDLC 000020RG   |
| BITD 000030R     | 002 PRIOR 000054R     | 002 ZMAN 000011R      | \$EXIT = 000000   | \$ZCOU = ***** GX  |
| CS.DLC= ***** GX | SETFLG 000170R        | ZMANL = 000007        | \$FAIL = 177777   | \$ZCOUL = ***** GX |
| CS.MOD= ***** GX | SETNAM 000132R        | \$ALPHA= 000022       | \$LAMBDA= 000000  | \$ZMFL = ***** GX  |
| C.FLG 000006     | SETNCT 000426R        | \$ANY = 000020        | \$MOVPT= ***** GX | \$ZMFL = ***** GX  |
| C.LEN = 000014   | SETNLT 000376R        | \$BLANK= 000006       | \$MOVST= ***** GX | \$ZTIM = ***** GX  |
| C.NAM 000004     | SETPRI 000350R        | \$BUFI = ***** GX     | \$MVASC= ***** GX | \$ZTIML= ***** GX  |
| C.NCT 000013     | TEMP 000000R          | \$CAT5 = ***** GX     | \$NUMBR= 000002   | \$ZFLG= 177777     |
| C.NLT 000012     | ZDLC 000002R          | \$CLNUP= ***** GX     | \$OBUF = ***** GX | \$ZKEY= 000005     |
| C.PRI 000010     | ZDLCL = 000007        | \$CNVD = ***** GX     | \$OFDB = ***** GX | \$ZSTA= 000000     |
| C.STS = ***** GX | ZF 000122R            | 002 \$CNVO = ***** GX | \$RAD50= 000016   | \$ZTMP= 000024R    |
| DLCKTB 000000RG  | 003 ZF.COUL= ***** GX | \$DIGIT= 000024       | \$STRNG= 000004   | .CONER= ***** GX   |
| DLCSTB 000000RG  | 002 ZF.MAN= ***** GX  | \$DNUMB= 000014       | \$SUBXP= 000010   | .PSTPT= ***** GX   |
| EXIT 000130R     | 002 ZF.MFL= ***** GX  | \$EOS = 000012        | \$TMLST= ***** GX | .TPARS= ***** GX   |
| F.NRBD= ***** GX |                       |                       |                   |                    |

. ABS. 000014 000 (RW,I,GBL,ABS,OVR)  
000456 001 (RW,I,LCL,REL,CON)  
\$STATE 000136 002 (RW,D,LCL,REL,CON)  
\$KTAB 000014 003 (RW,D,LCL,REL,CON)  
\$KSTR 000027 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10424 Words ( 41 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:30.45  
SY:UPDDL C.V2,C132,134JUPDDL C/CR/-SP=SY:[1,1]RSXMCM.SML/ML,C130,110JNETLIB/ML,C132,10JUPDDL C



UPDDSA      CREATED BY    MACRO    ON 29-JUN-85 AT 05:50      PAGE 2      J 5

MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|           |        |       |       |
|-----------|--------|-------|-------|
| EPRINT    | #4-79  |       |       |
| ERROR\$   | #4-68  |       |       |
| PRINT     | #4-63  |       |       |
| \$PTNMB   | #4-112 |       |       |
| \$PTNUM   | #4-111 | 6-177 | 6-178 |
| \$PTR50   | #4-116 | 6-176 | 6-179 |
| \$\$\$PTN | #4-91  | 4-111 | 4-112 |

.TITLE UPDDST - UPDATE DESTINATION DESCRIPTOR IN CETAB  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

UPDATE DESTINATION DESCRIPTOR IN CETAB

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/R SX V1.0

```

135 .SBTTL LOCAL DATA
136
137 :
138 : DEFINE THE DTE DESCRIPTOR BLOCK
139 :
140 000000 .ASECT
141 000000 .=0
142 000000 .BLKW 1
143 000002 .BLKW 1
144 000004 .BLKW 5
145 000016 .BLKW 1
146 000020 .BLKW 3
147 000026 .BLKW 1
148 000030 .BLKW 1
149 000032 .BLKW 1
150 000034 .BLKW 2
151 000040 .PSECT
152 000000 .NLIST BEX
153
154 :
155 000000 011 104 124 DTEDF: .ASCII <11>/DTE$DF/<11>
156 000010 DTEN=-DTEDF
157
158 000010 057760 ONSTR: .RAD50 /ON/
159 000012 057266 .RAD50 /OFF/
160
161 .LIST BEX
162 .EVEN
163

```

```

206 .SBTTL SETCLS - WRITE OUT EVENT CLASS
207
208 ;+
209 *** - SETCLS - WRITE OUT EVENT CLASS
210 ;
211 INPUT:
212 .PNUMB = THE CLASS NUMBER
213 .PNUMH = THE CLASS NUMBER (HIGH ORDER)
214 $OBUF = CURRENT POSITION IN OUTPUT LINE
215 TEMP = TEMPLATE ADDRESS
216 ;
217 OUTPUT:
218 C-BIT = SUCCESS/FAILURE
219 ;
220 000360 016702 177414 SETCLS: MOV TEMP,R2 ; GET ADDRESS OF EVENT TEMPLATE
221 000364 012767 000006 177410 MOV #C.EVT,EVTOFF ; SET UP EVENT MASK OFFSET
222 000372 016201 000004 MOV C.CLS(R2),R1 ; GET EVENT CLASS
223 000376 000411 BR SETNUM ; WRITE OUT NUMBER

```

141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168  
169  
170  
171  
172  
173  
174  
175  
176  
177  
178  
179

000160  
000160  
016700 000000G  
016702 000000G  
062502  
011202  
001403  
004567 000016  
000005  
0112720 000054  
010067 000000G  
000216  
000220 000205  
  
010201  
062501  
116103 177777  
001402  
112120  
077302  
000205  
  
000001

```
.SBTTL PUTFIL - WRITE OUT FILE SPEC
*
*** - PUTNOD - WRITE OUT FILE SPEC
:
: CALLING SEQUENCE:
:
: JSR R5,PUTFIL
: .WORD OFFSET
:
: OUTPUT:
: C-BIT = SUCCESS/FAILURE
:
: -
:
PUTFIL:
: SAVRG R3 ; SAVE REGISTER
: MOV $0BUF,R0 ; POINT AT CURRENT POSITION IN LINE
: .TEMP,R2 ; POINT AT TEMPLATE
: ADD (R5)+,R2 ; POINT TO FILE SPEC IN TEMPLATE
: MOV (R2),R2 ; POINT TO FILE DESCRIPTOR
: BEQ 10$; IF EQ, NO FILE
: JSR R5,20$; MOVE FILESPEC
: .WORD F.DEV
10$: MOV #<','>,(R0)+ ; INSERT COMMA
: MOV R0,$0BUF ; RESTORE BUFFER POINTER
: RESRG R3 ; RESTORE REGISTER
: RTS R5 ; DONE
:
: ; MOVE A FIELD
20$: MOV R2,R1 ; COPY ADDRESS OF FILE DESCRIPTOR
: ADD (R5)+,R1 ; POINT TO FIELD OF INTEREST
: MOV -1(R1),R3 ; GET FIELD LENGTH
: BEQ 40$; IF EQ, NO TEXT
30$: MOV (R1)+,(R0)+ ; ELSE MOVE IT
: SOB R3,30$; TO OUTPUT BUFFER
40$: RTS R5 ; DONE
:
: .END
```

UPDFEA      CREATED BY MACRO ON 29-JUN-85 AT 05:52      PAGE 2      J 10

MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |       |       |        |        |        |        |        |        |        |        |
|---------|-------|-------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$ | #5-95 |       |        |        |        |        |        |        |        |        |
| ISTAT\$ | #4-68 | 5-95  |        |        |        |        |        |        |        |        |
| MTRAN\$ | #5-95 |       |        |        |        |        |        |        |        |        |
| PRINT   | #4-59 |       |        |        |        |        |        |        |        |        |
| STAT\$  | #4-68 | 5-98  | #5-101 | #5-104 | #5-107 | #5-110 | #5-113 | #5-116 | #5-121 | #5-124 |
| TRAN\$  | #4-68 | #5-99 | #5-102 | #5-105 | #5-108 | #5-111 | #5-114 | #5-117 | #5-122 |        |

350  
 351  
 352  
 353  
 354  
 355  
 356  
 357  
 358  
 359  
 360  
 361

362 000560  
 363 000560 016700 177214  
 364 000564 005001  
 365 000566 156001 000014  
 366 000572 016700 000000G  
 367 000576 004767 000000G  
 368 000602 010067 000000G  
 369 000606 000207

.SBTTL SETNLT - SET NUMBER OF LINE TABLES TO ALLOCATE SPACE  
 \*\*\* - SETNLT - SET NUMBER OF LINE TABLES TO ALLOCATE SPACE  
 INPUTS:  
 TEMP = TEMPLATE ADDRESS  
 \$OBUF = CURRENT POSITION IN OUTPUT BUFFER  
 OUTPUTS:  
 -

SETNLT:  
 MOV TEMP,R0 ; GET THE TEMPLATE ADDRESS  
 CLR R1 ; GET THE NUMBER OF LINE TABLES  
 BISB C,NLT(R0),R1 ; ..WITHOUT SIGN EXTEND  
 MOV \$OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER  
 CALL \$CNVD ; CONVERT NUMBER TO DECIMAL  
 MOV R0,\$OBUF ; UPDATE THE POINTER INTO THE OUTPUT BUFFER  
 RETURN

```

129 .SBTTL $UPLOG - UPDATE LOGGING STATE TEMPLATE
130 ;+
131 ;*** - $UPLOG - UPDATE LOGGING STATE TEMPLATE
132 ; THIS ROUTINE IS CALLED AS AN ACTION ROUTINE BY TPARS.
133 ;
134 ; INPUT:
135 ; R3-R5 - TPARS REGISTERS
136 ;
137 ; OUTPUT:
138 ; C-BIT = SUCCESS/FAILURE
139 ;
140 ; -
141
142 $UPLOG::
143 000002 MOV #TMLST,RO ; POINT AT TEMPLATE LIST
144 000002 012700 000000G 10$: MOV RO,R1 ; SAVE CURRENT TEMPLATE
145 000006 010001 MOV (R0),RO ; GET NEXT TEMPLATE
146 000010 011000 BEQ 101$; IF EQ, END OF LIST
147 000012 001441 CMPB #CS.LOG,C.STS(R0) ; IS THIS THE LOGGING STATE TEMPLATE ?
148 000014 122760 000000G 000000G BNE 101$; IF NE, NO .. KEEP LOOKING
149 000022 001371
150 ;
151 ; ASSUME THIS IS THE RIGHT TEMPLATE
152 ;
153 000024 011011 MOV (R0),(R1) ; REMOVE FROM LIST
154 000026 010067 177746 MOV RO,NEXT ; SAVE TEMPLATE ADDRESS
155 000032 032760 000000G 000000G BIT #CS.MOD,C.STS(R0) ; WAS THIS TEMPLATE MODIFIED ?
156 000040 001004 BNE 15$; IF NE, YES .. UPDATE THE RECORD
157 000042 012767 000000G 000000C MOV #SBUF1,$OFDB+F.NRBD+2 ; OUTPUT THE INPUT BUFFER
158 000050 000421 BR 30$; AND LEAVE
159 000052 062767 000004 177720 15$: ADD #C.STA,NEXT ; POINT AT FIRST LOCATION IN TEMPLATE
160 000060 004767 000000G CALL $MOVST ; MOVE FIRST FIELD INTO OUTPUT BUFFER
161 000064 010546 MOV R5,-(SP) ; SAVE R5
162 000066 005001 CLR R1 ; IGNORE BLANKS
163 000070 012702 000000' MOV #LOGKTBL,R2 ; KEYWORD TABLE
164 000074 012705 000000' MOV #LOGSTBL,R5 ; STATE TABLE
165 000100 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
166 000104 103002 BCC 20$; NORMAL RETURN IF NO ERROR
167 000106 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
168 000112 012605 20$: MOV (SP)+,R5 ; RESTORE R5
169 000114 000207 30$: RETURN
170 ;
171 ; ERROR CONDITIONS
172 ;
173 000116 000167 000000G 101$: JMP .CONER ; CONSISTENCY ERROR

```



```

227
228
229
230
231
232
233
234
235
236
237
238
239 000156
240 000156 016700 000000G
241 000162 016701 177612
242 000166 062701 000010
243 000172 012102
244 000174 004767 000000G
245 000200 112720 000076
246 000204 010067 000000G
247 000210 000207

 .SBTTL SETNID - SET UP NODE ID
 ;
 ; *** - SETNID - SETUP NODE ID
 ;
 ; INPUT:
 ; TEMP = TEMPLATE ADDRESS
 ; (TEMP)+C.NID = NODE ID
 ;
 ; OUTPUT:
 ;
 ; -
 ;
 SETNID:
 MOV $OBUF,R0 ; POINT AT OUTPUT BUFFER
 MOV TEMP,R1 ; GET START ADDRESS OF NODE ID
 ADD #C.NID,R1 ;
 MOV (R1),R2 ; GET LENGTH OF NODE ID
 CALL $MVASC ; MOVE NODE ID INTO OUTPUT BUFFER
 MOVB #'>,(R0)+ ; INSERT DELIMETER
 MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
 RETURN

```

```

214 .SBTTL SETNAM - WRITE OUT OBJECT NAME
215 :+
216 :*** - SETNAM - WRITE OUT OBJECT NAME
217 :
218 :INPUT:
219 :.PSTPT = START ADDRESS OF THE OBJECT NAME IN ASCII
220 :.PSTCN = NUMBER OF CHARACTERS IN THE OBJECT NAME
221 :$OBJF = CURRENT POSITION IN OUTPUT BUFFER
222 :TEMP = TEMPLATE ADDRESS
223 :
224 :OUTPUT:
225 :THE RAD50 TASK NAME IS WRITTEN INTO THE OUTPUT BUFFER
226 :
227 :-
228
229 000356
230 000356 016702 177416
231 000362 005762 000006
232 000366 001003
233 000370 005762 000010
234 000374 001422
235
236 000376 016700 000000G
237 000402 112720 000074
238 000406 016201 000006
239 000412 004767 000000G
240 000416 016702 177336
241 000422 016201 000010
242 000426 004767 000000G
243 000432 112720 000076
244 000436 010067 000000G
245 000442 000207

 SETNAM:
 MOV TEMP,R2 ; GET ADDRESS OF TEMPLATE
 TST C.NAM(R2) ; IS NAME ALL BLANKS ?
 BNE 10$; IF NE, NO
 TST C.NAM+2(R2) ; STILL ALL BLANKS ?
 BEQ 20$; IF EQ, YES

 10$: MOV $OBJF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
 MOVB #'<,(R0)+ ; INSERT DELIMETER
 MOV C.NAM(R2),R1 ; GET RAD50 TASK NAME
 CALL $C5TA ; CONVERT TASK NAME TO ASCII
 MOV TEMP,R2 ; GET TEMPLATE ADDRESS .. AGAIN !!
 MOV C.NAM+2(R2),R1 ; GET 2ND HALF OF TASK NAME
 CALL $C5TA ; CONVERT TASK NAME TO ASCII
 MOVB #'>,(R0)+ ; INSERT DELIMETER
 MOV R0,$OBJF ; UPDATE POINTER INTO OUTPUT BUFFER

 20$: RETURN

```

143 000014  
144  
145 000014

TRANS\$ \$EOS,\$EXIT,\$CLNUP  
STATE\$

```

57 .SBTTL MACRO DEFINITIONS
58
59 ; LOCAL MACROS
60
61
62 .MACRO PRINT TEXT
63 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM
66
67 .MACRO ERROR$ TEXT
68
69 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
70 ; MESSAGE STRING.
71
72 .IF DIF <TEXT><R0>
73 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
74 .ENDC
75 CALL $CFERR ; PRINT ERROR MESSAGE
76 .ENDM ERROR$
77
78 .MACRO EPRINT TEXT
79
80 ; PRINT TEXT ON ERROR LUN
81
82 .IF DIF <TEXT><R0>
83 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
84 .ENDC
85 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
86 .ENDM EPRINT
87
88 ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
89
90 .MACRO $$$PTN ROU
91 .MACRO 'ROU OFS,OCTL,OPT,?L
92 .IF NB <OPT>
93 MOV OPT,R2
94 CALL $PTOPT
95 BCS L
96 .IFTF
97 MOV OFS,R2
98 .IF NB <OCTL>
99 MOV SP,R1
100 .IFF
101 CLR R1
102 .ENDC ; IF NB <OCTL>
103 CALL 'ROU
104 .IFT ; IF NB <OPT>
105
106 L:
107 .ENDC ; IF NB <OPT>
108 .ENDM 'ROU
109 .ENDM $$$PTN
110
111 $$$PTN $PTNUM
112 $$$PTN $PTNMB
113
114 ; PUT RAD50 TEMPLATE FIELD TO BUFFER

```

|         |          |        |          |         |         |          |         |           |          |        |       |              |          |       |
|---------|----------|--------|----------|---------|---------|----------|---------|-----------|----------|--------|-------|--------------|----------|-------|
| BITS    | 000016R  | 002    | SETNCT   | 000724R | ZF.MAN= | *****    | GX      | \$BUFI    | =        | *****  | GX    | \$RAD50=     | 000016   |       |
| BITO    | 000030R  | 002    | SETNLT   | 000674R | ZF.MFL= | *****    | GX      | \$CAT5    | =        | *****  | GX    | \$STRNG=     | 000004   |       |
| CS.DDM= | *****    | GX     | SETPRI   | 000646R | ZF.MUX= | *****    | GX      | \$CLNUP=  | *****    | GX     |       | \$SUBXP=     | 000010   |       |
| CS.MOD= | *****    | GX     | TEMP     | 000000R | ZF.PSE= | *****    | GX      | \$CNVD    | =        | *****  | GX    | \$TMLST=     | *****    | GX    |
| C.FLG   | 000006   |        | ZDDM     | 000002R | ZF.TJM= | *****    | GX      | \$CNVO    | =        | *****  | GX    | \$UPDDM      | 000102RG |       |
| C.LEN   | =        | 000014 | ZDDML    | =       | 000007  | ZKMX     | 000020R | \$DIGIT=  | 000024   |        |       | \$ZCOU       | =        | ***** |
| C.NAM   | 000004   |        | ZDIA     | 000054R | ZKMXL   | =        | 000007  | \$DNUMB=  | 000014   |        |       | \$ZCOUL=     | *****    | GX    |
| C.NCT   | 000013   |        | ZDIAL    | =       | 000007  | ZLMC     | 000036R | \$EOS     | =        | 000012 |       | \$ZMFL       | =        | ***** |
| C.NLT   | 000012   |        | ZDLC     | 000011R | ZLMCL   | =        | 000007  | \$ERROR=  | *****    | GX     |       | \$ZMFL=      | *****    | GX    |
| C.PRI   | 000010   |        | ZDLCL    | =       | 000007  | ZMAN     | 000072R | \$EXIT    | =        | 000000 |       | \$ZTIM       | =        | ***** |
| C.STS   | =        | *****  | ZDVP     | 000063R | ZMANL   | =        | 000007  | \$FAIL    | =        | 177777 |       | \$ZTIML=     | *****    | GX    |
| DDMKTB  | 000000RG | 003    | ZDVPL    | =       | 000007  | ZMUX     | 000027R | \$LAMBDA= | 000000   |        |       | \$\$\$\$FLG= | 177777   |       |
| DDMSTB  | 000000RG | 002    | ZF       | 000156R | 002     | ZMUXL    | =       | 000007    | \$MOVPT= | *****  | GX    | \$\$\$\$KEY= | 000014   |       |
| EXIT    | 000164R  | 002    | ZF.COUL= | *****   | GX      | ZPSE     | 000045R | \$MOVST=  | *****    | GX     |       | \$\$\$\$STA= | 000000   |       |
| F.NRBD= | *****    | GX     | ZF.DIA=  | *****   | GX      | ZPSEL    | =       | 000007    | \$MVASC= | *****  | GX    | \$\$\$\$TMP= | 000060R  | 004   |
| NCT     | 000140R  | 002    | ZF.DLC=  | *****   | GX      | \$ALPHA= | 000022  | \$NUMBR=  | 000002   |        |       | .CONER=      | *****    | GX    |
| PRIOR   | 000110R  | 002    | ZF.DVP=  | *****   | GX      | \$ANY    | =       | 000020    | \$OBUF   | =      | ***** | .PSTPT=      | *****    | GX    |
| SETFLG  | 000252R  |        | ZF.KMX=  | *****   | GX      | \$BLANK= | 000006  | \$OFDB    | =        | *****  | GX    | .TPARS=      | *****    | GX    |
| SETNAM  | 000214R  |        | ZF.LMC=  | *****   | GX      |          |         |           |          |        |       |              |          |       |

. ABS. 000014 000 (RW,I,GBL,ABS,OVR)  
 000754 001 (RW,I,LCL,REL,CON)  
 \$STATE 000172 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000032 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000063 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10556 Words ( 42 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:36.19  
 SY:UPDDDM.V2,[132,134]UPDDDM/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDDDM

```

222 .SBTTL SETINC - INCOMING TIMER
223 :+
224 *** - SETINC - SETUP INCOMING TIMER
225 :
226 INPUT:
227 TEMP = TEMPLATE ADDRESS
228 $OBUF = CURRENT POSITION IN OUTPUT BUFFER
229 :
230 OUTPUT:
231 :
232 :-
233
234 SETINC:
235 000170 016702 177604 MOV TEMP,R2 ; GET START OF TEMPLATE
236 000174 005001 CLR R1 ; GET INCOMING TIMER
237 000175 156201 000012 BISB C.INCT(R2),R1 ;
238 000202 016700 000000G MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
239 000206 004767 000000G CALL $CNVD ; CONVERT NUMBER TO DECIMAL
240 000212 010067 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
241 000216 000207 RETURN

```

\*\*FILE\*\*ID\*\*UPDDL

```

UU UU PPPPPPP DDDDDDDD DDDDDDDD LL CCCCCCCC
UU UU PPPPPPP DDDDDDDD DDDDDDDD LL CCCCCCCC
UU UU PP PP DD DD DD DD LL CC
UU UU PP PP DD DD DD DD LL CC
UU UU PP PP DD DD DD DD LL CC
UU UU PP PP DD DD DD DD LL CC
UU UU PPPPPPP DD DD DD DD LL CC
UU UU PPPPPPP DD DD DD DD LL CC
UU UU PP DD DD DD DD LL CC
UU UU PP DD DD DD DD LL CC
UU UU PP DD DD DD DD LL CC
UU UU PP DD DD DD DD LL CC
UUUUUUUU PP DDDDDDDD DDDDDDDD LLLLLLLLLL CCCCCCCC
UUUUUUUU PP DDDDDDDD DDDDDDDD LLLLLLLLLL CCCCCCCC

```

```

LL SSSSSSS TTTTTTTTT
LL SSSSSSS TTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLL SSSSSSS TT
LLLLLLLL SSSSSSS TT

```

SYMBOL CROSS REFERENCE

CREF 04.00

| SYMBOL  | VALUE       | REFERENCES                                              |
|---------|-------------|---------------------------------------------------------|
| CS.DLC  | = ***** GX  | 6-178                                                   |
| CS.MOD  | = ***** GX  | 6-185                                                   |
| C.FLG   | = 000006    | #4-88 8-247 8-252 8-257 8-262                           |
| C.LEN   | = 000014    | #4-92                                                   |
| C.NAM   | = 000004    | #4-87 7-223                                             |
| C.NCT   | = 000013    | #4-91 11-329                                            |
| C.NLT   | = 000012    | #4-90 10-308                                            |
| C.PRI   | = 000010    | #4-89 9-286                                             |
| C.STS   | = ***** GX  | 6-178 6-185                                             |
| DLC.KTB | = 000000 RG | #5-100 6-192                                            |
| DLC.STB | = 000000 RG | #5-100 6-193                                            |
| F.NRBD  | = ***** GX  | *6-187                                                  |
| SET.FLG | = 000170 R  | #8-240                                                  |
| SET.NAM | = 000132 R  | #7-217                                                  |
| SET.NCT | = 000426 R  | #11-326                                                 |
| SET.NLT | = 000376 R  | #10-305                                                 |
| SET.PRI | = 000350 R  | #9-284                                                  |
| TEMP    | = 000000 R  | #4-73 *6-184 7-222 8-242 9-285 10-306 11-327            |
| ZDLC    | = 000002 R  | #4-74 8-244                                             |
| ZDLC.L  | = 000007    | #4-75 8-245                                             |
| ZF.COJ  | = ***** GX  | 8-257                                                   |
| ZF.MAN  | = ***** GX  | 8-262                                                   |
| ZF.MFL  | = ***** GX  | 8-252                                                   |
| ZF.TIM  | = ***** GX  | 8-247                                                   |
| ZMAN    | = 000011 R  | #4-76 4-77 8-264                                        |
| ZMANL   | = 000007    | #4-77 8-265                                             |
| \$ALPHA | = 000022    | #5-100                                                  |
| \$ANY   | = 000020    | #5-100                                                  |
| \$BLANK | = 000006    | #5-100                                                  |
| \$BUF1  | = ***** GX  | 6-187                                                   |
| \$CAT5  | = ***** GX  | 7-221                                                   |
| \$CNVD  | = ***** GX  | 10-310 11-331                                           |
| \$CNVO  | = ***** GX  | 9-288                                                   |
| \$DIGIT | = 000024    | #5-100                                                  |
| \$DNUMB | = 000014    | #5-100                                                  |
| \$EOS   | = 000012    | #5-100                                                  |
| \$ERROR | = ***** GX  | *6-196                                                  |
| \$EXIT  | = 000000    | #5-100                                                  |
| \$FAIL  | = 177777    | #5-100                                                  |
| \$GPRM  | = *****     | 5-100                                                   |
| \$LAMDA | = 000000    | #5-100                                                  |
| \$MOVPT | = ***** GX  | 7-218                                                   |
| \$MOVST | = ***** GX  | 6-189                                                   |
| \$MVASC | = ***** GX  | 8-246 8-251 8-256 8-261 8-266                           |
| \$NUMBR | = 000002    | #5-100                                                  |
| \$OBUF  | = ***** GX  | 8-243 *8-268 9-287 *9-289 10-309 *10-311 11-330 *11-332 |
| \$OFDE  | = ***** GX  | *6-187                                                  |
| \$RAD50 | = 000016    | #5-100                                                  |
| \$ONLY  | = *****     | 5-100 5-100                                             |
| \$STRNG | = 000004    | #5-100                                                  |
| \$SUBXP | = 000010    | #5-100                                                  |
| \$TMLST | = ***** GX  | 6-174                                                   |



\*\*FILE\*\*ID\*\*UPDDSC

```

UU UU PPPPPPP DDDDDDDD DDDDDDDD SSSSSSSS CCCCCCCC
UU UU PPPPPPP DDDDDDDD DDDDDDDD SSSSSSSS CCCCCCCC
UU UU PP PP DD DD DD DD SS CC
UU UU PP PP DD DD DD DD SS CC
UU UU PP PP DD DD DD DD SS CC
UU UU PP PP DD DD DD DD SS CC
UU UU PPPPPPP DD DD DD DD SSSSSS CC
UU UU PPPPPPP DD DD DD DD SSSSSS CC
UU UU PP DD DD DD DD SS CC
UU UU PP DD DD DD DD SS CC
UU UU PP DD DD DD DD SS CC
UU UU PP DD DD DD DD SS CC
UUUUUUUU PP DDDDDDDD DDDDDDDD SSSSSSSS CCCCCCCC
UUUUUUUU PP DDDDDDDD DDDDDDDD SSSSSSSS CCCCCCCC

```

```

....
....
....

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLL SSSSSSSS TT
LLLLLLLL SSSSSSSS TT

```

```

57 .SBTTL MACRO CALLS AND LOCAL DATA
58 .SBTTL MACRO DEFINITIONS
59
60 ; LOCAL MACROS
61 ;
62
63 .MACRO PRINT TEXT
64 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
65 CALL $PRINT ; PRINT MESSAGE
66 .ENDM PRINT
67
68 .MACRO ERRORS$ TEXT
69 ;
70 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
71 ; MESSAGE STRING.
72 ;
73 .IF DIF <TEXT><RO>
74 MOV TEXT,RO ; GET ADDRESS OF ERROR MESSAGE
75 .ENDC
76 CALL $CFERR ; PRINT ERROR MESSAGE
77 .ENDM ERRORS$
78
79 .MACRO EPRINT TEXT
80 ;
81 ; PRINT TEXT ON ERROR LUN
82 ;
83 .IF DIF <TEXT><RO>
84 MOV TEXT,RO ; GET ADDRESS OF ASCIZ STRING TO PRINT
85 .ENDC
86 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
87 .ENDM EPRINT
88
89 ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
90 ;
91 .MACRO $$$PTN ROU
92 .MACRO 'ROU OFS,OCTL,OPT,?L
93 .IF NB <OPT>
94 MOV OPT,R2
95 CALL $PTOPT
96 BCS L
97 .IFTF
98 MOV OFS,R2
99 .IF NB <OCTL>
100 MOV SP,R1
101 .IFF
102 CLR R1
103 .ENDC ; IF NB <OCTL>
104 CALL 'ROU
105 .IFT ; IF NB <OPT>
106
107 .ENDC ; IF NB <OPT>
108 .ENDM ; 'ROU
109 .ENDM ; $$$PTN
110
111 $$$PTN $PTNUM
112 $$$PTN $PTNMB
113 ;

```

```

165 .SBTTL $UPDTE - UPDATE DTE DEFINITION
166 :+
167 :*** - $UPDTE - UPDATE DTE DEFINITION
168 :
169 :INPUT:
170 :R5 - TEMPLATE ADDRESS
171 :
172 :OUTPUT:
173 :C-BIT = SUCCESS/FAILURE
174 :
175 :-
176
177 000014 $UPDTE::
178 000014 012767 000000G 000000C MOV #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
179 000022 010567 000000G MOV R5,TEMP ; UPDATE SCANNED POINTER
180 000026 005725 TST (R5)+ ; POINT TO C.STS
181 000030 010567 000000G MOV R5,.FLAGS ; SAVE ADDRESS OF FLAGS WORD
182 000034 012767 177777 000000G MOV #-1,COMMA ; INITIALIZE FOR OPTIONS
183 000042 012700 000000G MOV #.BUFF,R0 ; POINT AT START OF OUTPUT BUFFER
184 000046 012701 000000' MOV #DTEDF,R1 ; POINT AT START OF DTE DEFINITION
185 000052 012702 000010 MOV #DTELN,R2 ; GET LENGTH OF DTE DEFINITION
186 000056 004767 000000G CALL $MVASC ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
187 000062 010067 000000G MOV R0,$OBUF ; SAVE CURRENT POSITION IN OUTPUT BUFFER
188 000066 $PTR50 #C.ADR,#5 ; ADDRESS
189 000102 004767 000152 CALL SETLIN ; LINE
190 000106 $PTNUM #C.HSH ; HASH
191 000120 $PTNUM #C.CT ; COUNTER TIMER
192 000132 004767 000056 CALL SETSTA ; STATE
193 000136 $PTR50 #C.NET,#2,#CS.NET ; NETWORK NAME
194 000164 016701 000000G MOV $OBUF,R1 ; CALCULATE LENGTH OF BUFFER
195 000170 162701 000000G SUB #.BUFF,R1 ;
196 000174 010167 000000C MOV R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
197 000200 004767 000000G CALL $WDATA ; UPDATE RECORD
198 000204 012767 000000G 000000C MOV # $BUF1,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE "END$DF" LINE
199 000212 000207 RETURN
200

```

225  
 226  
 227  
 228  
 229  
 230  
 231  
 232  
 233  
 234  
 235  
 236  
 237  
 238  
 239  
 240  
 241  
 242  
 243  
 244

.SBTTL SETMSK - WRITE OUT EVENT MASK

\*\*\* - SETMSK - WRITE OUT EVENT MASK

INPUT:

.PNUMB = THE EVENT MASK NUMBER  
 .PNUMH = THE EVENT MASK NUMBER (HIGH ORDER)  
 \$OBUF = CURRENT POSITION IN OUTPUT LINE  
 TEMP = TEMPLATE ADDRESS

OUTPUT:

C-BIT = SUCCESS/FAILURE

SETMSK: MOV TEMP,R2 ; GET ADDRESS OF EVENT TEMPLATE  
 ADD EVTOFF,R2 ; POINT TO EVENT MASKS  
 MOV (R2),R1 ; GET EVENT MASK  
 ADD #2,EVTOFF ; SAVE UPDATED EVENT MASK OFFSET  
 BR SETNUM ; WRITE OUT EVENT MASK

177374  
 177372  
 011201  
 062767 000002 177362  
 000400

UPDFIL - CFE UPDATE REMOTE CHAR MACRO V05.03b Saturday 29-Jun-85 05:52 <sup>K 9</sup> Page 6-1  
Symbol table

|        |          |                 |                  |                   |                   |
|--------|----------|-----------------|------------------|-------------------|-------------------|
| FILDF  | 000000R  | F\$LEN = 000016 | F.LEN = 000043   | \$MVASC= ***** GX | \$WDATA= ***** GX |
| FILLN  | = 000010 | F\$SEC 000004   | F.LNK 000000     | \$OBUF = ***** GX | .BUFF = ***** GX  |
| F\$DIA | 000012   | F\$TER 000006   | F.NRBD= ***** GX | \$OFDB = ***** GX | .FLAGS= ***** GX  |
| F\$DUM | 000014   | F.CNT 000002    | MXFSP = 000036   | \$SAVRG= ***** GX | .TEMP = ***** GX  |
| F\$LD  | 000010   | F.DEV 000005    | PUTFIL 000160R   | \$UPFIL 000010RG  |                   |

. ABS. 000043 000 (RW,I,GBL,ABS,OVR)  
000242 001 (RW,I,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 218 Words ( 1 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:05.37  
SY:UPDFIL.V2,[132,134]UPDFIL/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDFIL

\*\*FILE\*\*ID\*\*UPDLLC

```

UU UU PPPPPPP DDDDDDDD LL LL CCCCCCCC
UU UU PPPPPPP DDDDDDDD LL LL CCCCCCCC
UU UU PP PP DD DD LL LL CC
UU UU PP PP DD DD LL LL CC
UU UU PP PP DD DD LL LL CC
UU UU PP PP DD DD LL LL CC
UU UU PPPPPPP DD DD LL LL CC
UU UU PPPPPPP DD DD LL LL CC
UU UU PP DD DD LL LL CC
UU UU PP DD DD LL LL CC
UU UU PP DD DD LL LL CC
UU UU PP DD DD LL LL CC
UUUUUUUU PP DDDDDDDD LLLLLLLLLL LLLLLLLLLL CCCCCCCC
UUUUUUUU PP DDDDDDDD LLLLLLLLLL LLLLLLLLLL CCCCCCCC

```

```

LL SSSSSSSS TTTTTTTTTT
LL SSSSSSSS TTTTTTTTTT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LL SSSSSS TT
LL SSSSSS TT
LL SS TT
LL SS TT
LL SS TT
LL SS TT
LLLLLLLLLL SSSSSSSS TT
LLLLLLLLLL SSSSSSSS TT

```

```

371 .SBTTL SETEXT - SET THE PROCESS EXTENSION SIZE
372 ;
373 ; *** - SETEXT - SET THE PROCESS EXTENSION SIZE
374 ;
375 INPUTS:
376 TEMP = TEMPLATE ADDRESS
377 $OBUF = CURRENT POSITION IN OUTPUT BUFFER
378 ;
379 OUTPUTS:
380 ;
381 ; -
382 ;
383 SETEXT:
384 MOV TEMP,R0 ; GET TEMPLATE ADDRESS
385 MOVB C.EXT(R0),R1 ; GET THE PROCESS EXTENSION SIZE
386 MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
387 CALL $CNVD ; CONVERT NUMBER TO DECIMAL
388 MOV RO,$OBUF ; UPDATE POSITION IN OUTPUT BUFFER
389 RETURN
000610
000610 016700 177164
000614 116001 000015
000620 016700 000000G
000624 004767 000000G
000630 010067 000000G
000634 000207

```

```

175
176
177
178
179
180
181
182
183
184
185
186
187
188
189 000122
190 000122 016700 000000G
191 000126 016702 177646
192 000132 112201
193 000134 010267 177640
194 000140 004767 000000G
195 000144 010067 000000G
196 000150 000207
197
198
199 000001

.SBTTL SETNUM - SET THE NEXT NUMBER
*** - SETNUM - SET UP THE NEXT NUMBER
INPUT:
$OBUF = CURRENT POSITION IN THE OUTPUT BUFFER
NEXT = ADDRESS OF NEXT LOCATION IN THE TEMPLATE FILE
OUTPUT:
C-BIT = SUCCESS/FAILURE
$OBUF = NEXT LOCATION IN OUTPUT BUFFER
-
SETNUM:
MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
MOV NEXT,R2 ; GET ADDRESS OF NEXT LOCATION IN TEMPLATE
MOVB (R2)+,R1 ; GET NEXT VALUE
MOV R2,NEXT ; SAVE NEXT LOCATION IN TEMPLATE
CALL $CNVO ; CONVERT NUMBER TO DECIMAL
MOV R0,$OBUF ; SAVE CURRENT POSITION IN BUFFER
RETURN ; RETURN TO CALLER

.END

```



```

249 .SBTTL SETNNA - SET EXEC AREA
250 ;+
251 *** - SETNNA - SET EXEC AREA
252 ;
253 INPUT:
254 TEMP = TEMPLATE ADDRESS
255 (TEMP)+C.HQS = NODE ADDRESS
256 ;
257 OUTPUT:
258 ;
259 ;-
260
261 SETNNA:
262 000212 016700 000000G MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
263 000216 016702 177556 MOV TEMP,R2 ; POINT R2 AT TEMPLATE
264 000222 032762 000000G 000000G BIT #NO$EXA,C.STS(R2) ; EXEC AREA PARSED ?
265 000230 001413 BEQ 10$; NOPE
266 000232 016201 000052 MOV C.NUM(R2),R1 ; GET NODE ADDRESS
267 000236 042701 001777 BIC #1777,R1 ; ONLY AREA
268 000242 000301 SWAB R1 ; FORMAT IT
269 000244 006201 ASR R1 ; ...
270 000246 006201 ASR R1 ; ...
271 000250 004767 000000G CALL $CNVD ; CONVERT BINARY TO DECIMAL
272 000254 010067 000000G MOV RO,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
273 000260 000207 10$: RETURN
274
275 .SBTTL SETNNM - SET NODE NUMBER
276 ;+
277 *** - SETNNM - SET NODE NUMBER
278 ;
279 INPUT:
280 TEMP = TEMPLATE ADDRESS
281 (TEMP)+C.NUM = NODE NUMBER
282 ;
283 OUTPUT:
284 ;
285 ;-
286
287 SETNNM:
288 000262 016700 000000G MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
289 000266 016702 177506 MOV TEMP,R2 ; POINT R2 AT TEMPLATE
290 000272 016201 000052 MOV C.NUM(R2),R1 ; GET NODE NUMBER
291 000276 042701 176000 BIC #C1777,R1 ; CLEAR AREA
292 000302 004767 000000G CALL $CNVD ; CONVERT BINARY TO DECIMAL
293 000306 010067 000000G MOV RO,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
294 000312 000207 RETURN

```

```

247 .SBTTL SETVfy - SETUP OBJECT VERIFICATION LEVEL
248 ;+
249 ;*** - SETVfy - WRITE OUT OBJECT VERIFICATION LEVEL
250 ;
251 ; INPUT:
252 ; .PNUMB = THE OBJECT VERIFICATION LEVEL
253 ; .PNUMH = THE OBJECT VERIFICATION LEVEL (HIGH ORDER)
254 ; $OBUF = CURRENT POSITION IN OUTPUT LINE
255 ; TEMP = TEMPLATE ADDRESS
256 ;
257 ; OUTPUT:
258 ; THE VERIFICATION LEVEL IS WRITTEN TO THE OUTPUT LINE
259 ;
260 ; -
261
262 000444 SETVfy:
263 000444 016700 000000G MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT LINE
264 000450 016702 177324 MOV TEMP,R2 ; POINT AT TEMPLATE
265 000454 005001 CLR R1 ; GET THE VERIFICATION LEVEL
266 000456 156201 000005 BISB C.FLG(R2),R1 ; ..WITHOUT SIGN EXTEND
267 000462 042701 177770 BIC #^C<7>,R1 ; CLEAR ALL BUT THE VERIFICATION LEVEL
268 000466 004767 000000G CALL $CNVO ; CONVERT BINARY TO OCTAL
269 000472 010067 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT LINE
270 000476 000207 RETURN

```

```

147 .SBTTL $UPPAR - UPDATE PARTITION DEFINITION
148 +
149 *** - $UPPAR - UPDATE PARTITION DEFINITION
150 :
151 INPUT:
152 R3-R5 - TPARS REGISTERS
153 :
154 OUTPUT:
155 C-BIT = SUCCESS/FAILURE
156 :
157 -
158
159 000014 $UPPAR::
160 000014 012700 000000G MOV #STM.LST,k0 ; POINT AT START OF TEMPLATES
161 000020 010001 10$: MOV R0,R1 ; SAVE CURRENT TEMPLATE
162 000022 011000 MOV (R0),R0 ; GET NEXT TEMPLATE
163 000024 001426 BEQ 101$; IF EQ, END OF TEMPLATES
164 000026 122760 000000G 000000G CMPB #CS.PAR,C.STS(R0) ; IS THIS A PARTITION TEMPLATE ?
165 000034 001371 BNE 10$; IF NE, NO .. KEEP LOOKING
166 :
167 ; ASSUME THIS IS THE RIGHT TEMPLATE
168 :
169 000036 011011 MOV (R0),(R1) ; REMOVE TEMPLATE FROM THE CHAIN
170 000040 010067 177734 MOV R0,TEMP ; SAVE TEMPLATE ADDRESS
171 000044 004767 000000G CALL $MOVST ; MOVE FIRST FIELD INTO OUTPUT BUFFER
172 000050 010546 MOV R5,-(SP) ; SAVE R5
173 000052 005001 CLR R1 ; IGNORE BLANKS
174 000054 012702 000000' MOV #PARKTB,R2 ; KEYWORD TABLE
175 000060 012705 000000' MOV #PARSTB,R5 ; STATE TABLE
176 000064 004767 000000G CALL TPARS ; PARSE THE REST OF THE LINE
177 000070 103002 BCC 20$; NORMAL RETURN IF NO ERROR
178 000072 005267 000000G INC $ERROR ; INDICATE SYNTAX ERROR
179 000076 012605 20$: MOV (SP)+,R5 ; RESTORE R5
180 000100 000207 30$: RETURN
181 :
182 ; ERROR CONDITIONS
183 :
184 000102 000167 000000G 101$: JMP .CONER ; CONSISTENCY ERROR

```

```
114
115
116
117
118
119
120
```

```
;
MACRO $PTR50 OFS,WC
MOV OFS,R2
MOV WC,R1
CALL $PTR50
.ENDM
```

UPDDDM CREATED BY MACRO ON 29-JUN-85 AT 05:49 PAGE 1 L 1  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL  | VALUE      | REFERENCES                                             |
|---------|------------|--------------------------------------------------------|
| CS.DDM  | = ***** GX | 6-205                                                  |
| CS.MOD  | = ***** GX | 6-212                                                  |
| C.FLG   | 000006     | #4-109 8-274 8-275 8-276 8-277 8-278 8-279 8-280 8-281 |
|         |            | 8-282 8-283 8-284                                      |
| C.LEN   | = 000014   | #4-113                                                 |
| C.NAM   | 000004     | #4-108 7-250                                           |
| C.NCT   | 000013     | #4-112 11-347                                          |
| C.NLT   | 000012     | #4-111 10-326                                          |
| C.PRI   | 000010     | #4-110 9-304                                           |
| C.STS   | = ***** GX | 6-205 6-212                                            |
| DDMKTB  | 000000 RG  | #5-120 6-219                                           |
| DDMSTB  | 000000 RG  | #5-120 6-220                                           |
| F.NRBD  | = ***** GX | *6-214                                                 |
| SETFLG  | 000252 R   | #8-267                                                 |
| SETNAM  | 000214 R   | #7-244                                                 |
| SETNCT  | 000724 R   | #11-344                                                |
| SETNLT  | 000674 R   | #10-323                                                |
| SETPRI  | 000646 R   | #9-302                                                 |
| TEMP    | 000000 R   | #4-89 *6-211 7-249 8-269 9-307 10-324 11-345           |
| ZDDM    | 000002 R   | #4-90 4-90 8-271                                       |
| ZDDML   | = 000007   | #4-90 8-272                                            |
| ZDIA    | 000054 R   | #4-96 4-96 8-282                                       |
| ZDIAL   | = 000007   | #4-96 8-282                                            |
| ZDLC    | 000011 R   | #4-91 4-91 8-276                                       |
| ZDLCL   | = 000007   | #4-91 8-276                                            |
| ZDVP    | 000063 R   | #4-97 4-97 8-283                                       |
| ZDVPL   | = 000007   | #4-97 8-283                                            |
| ZF.COQ  | = ***** GX | 8-277                                                  |
| ZF.DIA  | = ***** GX | 8-282                                                  |
| ZF.DLC  | = ***** GX | 8-276                                                  |
| ZF.DVP  | = ***** GX | 8-283                                                  |
| ZF.KMX  | = ***** GX | 8-278                                                  |
| ZF.LMC  | = ***** GX | 8-279                                                  |
| ZF.MAN  | = ***** GX | 8-284                                                  |
| ZF.MFL  | = ***** GX | 8-275                                                  |
| ZF.MUX  | = ***** GX | 8-280                                                  |
| ZF.PSE  | = ***** GX | 8-281                                                  |
| ZF.TIM  | = ***** GX | 8-274                                                  |
| ZKMX    | 000020 R   | #4-92 4-92 8-278                                       |
| ZKMXL   | = 000007   | #4-92 8-278                                            |
| ZLMC    | 000076 R   | #4-94 4-94 8-279                                       |
| ZLMCL   | = 000007   | #4-94 8-279                                            |
| ZMAN    | 000072 R   | #4-98 4-98 8-284                                       |
| ZMANL   | = 000007   | #4-98 8-284                                            |
| ZMUX    | 000027 R   | #4-93 4-93 8-280                                       |
| ZMUYL   | = 000007   | #4-93 8-280                                            |
| ZPSE    | 000045 R   | #4-95 4-95 8-281                                       |
| ZPSEL   | = 000007   | #4-95 8-281                                            |
| \$ALPHA | = 000022   | #5-120                                                 |
| \$ANY   | = 000020   | #5-120                                                 |
| \$BLANK | = 000006   | #5-120                                                 |
| \$EUF   | = ***** GX | 6-214                                                  |

UPDDDM CREATED BY MACRO ON 29-JUN-85 AT 05:49 PAGE 2 M 1  
 SYMBOL CROSS REFERENCE CREF 04.00

```

243 .SBTTL SETOUT - OUTGOING TIMER
244
245 *** - SETOUT - SETUP OUTGOING TIMER
246
247 INPUT:
248 TEMP = TEMPLATE ADDRESS
249 $OBUF = CURRENT POSITION IN OUTPUT BUFFER
250
251 OUTPUT:
252
253
254
255 SETOUT:
256 MOV TEMP,R2 ; GET START OF TEMPLATE
257 CLR R1 ; GET OUTGOING TIMER
258 B1SB C.OUTT(R2),R1 ;
259 MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
260 CALL $CNVD ; CONVERT NUMBER TO DECIMAL
261 MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
262 RETURN

```

UPDDL - CFE UPDATE DLC PROCESS MACRO V05.03b Saturday 29-Jun-85 05:50 <sup>L 3</sup>  
Table of contents

|     |     |                                            |
|-----|-----|--------------------------------------------|
| 6-  | 161 | \$UPDL - UPDATE DLC DEFINITION             |
| 7-  | 204 | SETNAM - CHECK PROCESS NAME                |
| 8-  | 228 | SETFLG - SETUP THE FLAGS WORD              |
| 9-  | 272 | SETPRI - SET THE PRIORITY                  |
| 10- | 292 | SETNLT - STORE NUMBER OF LINE TABLES       |
| 11- | 314 | SETNCT - STORE NUMBER OF CONTROLLER TABLES |

SYMBOL CROSS REFERENCE

CREF    04.00

SYMBOL    VALUE                    REFERENCES

|           |          |    |        |
|-----------|----------|----|--------|
| \$UPDL C  | = 000020 | RG | #6-173 |
| \$ZCOU    | = *****  | GX | 8-259  |
| \$ZCOUL   | = *****  | GX | 8-260  |
| \$ZMFL    | = *****  | GX | 8-254  |
| \$ZMFL L  | = *****  | GX | 8-255  |
| \$ZT IM   | = *****  | GX | 8-249  |
| \$ZT IML  | = *****  | GX | 8-250  |
| \$\$\$FLG | = 177777 |    | #5-100 |
| \$\$\$KEY | = 177777 |    | #5-100 |
| .CONER    | = *****  | GX | 6-202  |
| .PSTPT    | = *****  | GX | 7-219  |
| .TPARS    | = *****  | GX | 6-194  |



UPDDSC - UPDATE DESTINATION CAL MACRO V05.03b Saturday 29-Jun-85 <sup>L 5</sup> 05:50  
Table of contents

|    |     |                                              |
|----|-----|----------------------------------------------|
| 4- | 57  | MACRO CALLS AND LOCAL DATA                   |
| 4- | 58  | MACRO DEFINITIONS                            |
| 5- | 131 | LOCAL DATA                                   |
| 6- | 151 | \$UPDSC - UPDATE DESTINATION CALL PARAMETERS |

```

114 ; PUT RAD50 TEMPLATE FIELD TO BUFFER
115 ;
116 .MACRO $PTR50 OFS,WC,OPT,?L
117 .IF NB <OPT>
118 MOV OPT,R2
119 CALL $PTOPT
120 BCS L
121 .IFTF
122 MOV OFS,R2
123 MOV WC,R1
124 CALL $PTR50
125 .IFT
126 L:
127 .ENDC
128 .ENDM
129

```

```

202 .SBTTL SETSTA - SET STATE
203 .SBTTL SETLIN - SET LINE ID
204 +
205 ** SETSTA - SET STATE
206 ** SETLIN - SET LINE ID
207 :
208 INPUT: CURRENT TEMPLATE.
209 :
210 OUTPUT: VALUE LOADED INTO OUTPUT BUFFER.
211 :
212 -
213
214 000214 016701 000000G SETSTA: MOV .TEMP,R1 ; POINT TO TEMPLATE
215 000220 010146 MOV R1,-($P) ; SAVE TEMPLATE ADDRESS
216 000222 005002 CLR R2 ; ASSUME STATE = ON
217 000224 105761 000026 TSTB C,STA(R1) ; CHECK STATE
218 000230 001001 BNE 10$; IF NO, ON
219 000232 122222 CMPB (R2)+,(R2)+ ; ELSE STATE = OFF
220 000234 012767 000010' 000000G 10$: MOV #ONSTR,.TEMP ; POINT TO START OF STRINGS
221 000242 012701 000001 MOV #1,R1 ; ONE RAD50 WORD
222 000246 004767 000000G CALL $PTR50 ; OUTPUT RAD50
223 000252 012667 000000G MOV (SP)+,.TEMP ; RESTORE TEMPLATE ADDRESS
224 000256 000207 RETURN
225
226 000260 SETLIN: $PTR50 #C.LIN,#1 ; RAD50 DDM NAME
227 000274 112777 000055 000000G MOVB #'-.@,$0BUF ; SEPARATOR
228 000302 005267 000000G INC $0BUF ; ADVANCE BUFFER POINTER
229 000306 012767 177777 000000G MOV #-1,.,COMMA ; SUPPRESS COMMAS
230 000314 $PTNUM #C.LIN+2,OCTAL ; CONTROLLER NUMBER
231 000326 032777 000000G 000000G BIT #CS.UNI,@.FLAGS ; CHECK OPTION
232 000334 001415 BEQ 10$; IF EQ, NOT SELECTED
233 000336 112777 000055 000000G MOVB #'-.@,$0BUF ; ELSE BUFFER CHAR
234 000344 005267 000000G INC $0BUF ; BUMP BUFPTR
235 000350 012767 177777 000000G MOV #-1,.,COMMA ; SUPPRESS COMMAS
236 000356 $PTNUM #C.LIN+4,OCTAL ; OUTPUT VALUE
237 000370 032777 000000G 000000G 10$: BIT #CS.TRB,@.FLAGS ; CHECK OPTION
238 000376 001415 BEQ 20$; IF EQ, NOT SELECTED
239 000400 112777 000056 000000G MOVB #'-.@,$0BUF ; ELSE BUFFER CHAR
240 000406 005267 000000G INC $0BUF ; BUMP BUFPTR
241 000412 012767 177777 000000G MOV #-1,.,COMMA ; SUPPRESS COMMAS
242 000420 $PTNUM #C.LIN+6,OCTAL ; OUTPUT VALUE
243 000432 000207 20$: RETURN
244
245 000001 .END

```

```

246 .SBTTL SETNUM - WRITE OUT NUMBER
247 :+
248 :*** - SETNUM - WRITE OUT NUMBER
249 :
250 :INPUT:
251 :R1 - NUMBER TO CONVERT
252 :
253 :OUTPUT:
254 :C-BIT = SUCCESS/FAILURE
255 :
256 :
257 :
258 SETNUM:
259 000422 016700 000000G MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN LINE
260 000426 004767 000000G CALL $CNVO ; CONVERT BINARY TO OCTAL
261 000432 010067 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT LINE
262 000436 000207

```

UPDFIL      CREATED BY MACRO ON 29-JUN-85 AT 05:52      PAGE 1      L 9

SYMBOL CROSS REFERENCE      CREF      04.00

| SYMBOL  | VALUE      | REFERENCES                                                    |
|---------|------------|---------------------------------------------------------------|
| FILDF   | 000000 R   | #4-54      4-55      5-110                                    |
| FILLN   | = 000010   | #4-55      5-111                                              |
| F\$DIA  | 000012     | #4-69      5-125                                              |
| F\$DUM  | 000014     | #4-70      5-128                                              |
| F\$LD   | 000010     | #4-68      5-122                                              |
| F\$LEN  | = 000016   | #4-71                                                         |
| F\$SEC  | 000004     | #4-66      5-116                                              |
| F\$TER  | 000006     | #4-67      5-119                                              |
| F.CNT   | 000002     | #4-83                                                         |
| F.DEV   | 000005     | #4-85      6-163                                              |
| F.LEN   | = 000043   | #4-86                                                         |
| F.LNK   | 000000     | #4-82                                                         |
| F.NRBD  | = ***** GX | 5-104      *5-105      *5-136      *5-138                     |
| MXFSP   | = 000036   | #4-78      4-85                                               |
| PUTFIL  | 000160 R   | 5-115      5-118      5-121      5-124      5-127      #6-155 |
| \$MVASC | = ***** GX | 5-112                                                         |
| \$OBUF  | = ***** GX | *5-113      5-130                                             |
| \$OFDB  | = ***** GX | 5-104      *5-105      *5-136      *6-165      *5-138         |
| \$SAVRG | = ***** GX | 5-103                                                         |
| \$UPFIL | 000010 RG  | #5-102                                                        |
| \$WDATA | = ***** GX | 5-137                                                         |
| .BUFF   | = ***** GX | 5-105      5-109      5-135                                   |
| .FLAGS  | = ***** GX | *5-108                                                        |
| .TEMP   | = ***** GX | *5-106      6-158                                             |

UPDFIL      CREATED BY MACRO ON 29-JUN-85 AT 05:52      PAGE 2      M 9

MACRO CROSS REFERENCE      CREF      04.00

UPDLLC - CFE UPDATE LLC PROCESS MACRO V05.03b Saturday 29-Jun-85 05:52 L 10  
Table of contents

|     |     |                                                      |
|-----|-----|------------------------------------------------------|
| 6-  | 183 | \$UPLLC - UPDATE LLC DEFINITION                      |
| 7-  | 226 | SETNAM - CHECK IF LLC PROCESS NAMES MATCH            |
| 8-  | 250 | SETFLG - SETUP THE FLAGS WORD                        |
| 9-  | 308 | SETPRI - SET THE PRIORITY                            |
| 10- | 329 | SETLIN - SET LINE NUMBER                             |
| 11- | 350 | SETNLT - SET NUMBER OF LINE TABLES TO ALLOCATE SPACE |
| 12- | 371 | SETEXT - SET THE PROCESS EXTENSION SIZE              |
| 13- | 391 | SETCTM - COUNTER TIMER                               |

391  
392  
393  
394  
395  
396  
397  
398  
399  
400  
401  
402  
403  
404  
405  
406  
407  
408  
409  
410  
411

000636  
000636 016702 177136  
000642 016201 000016  
000646 016700 000000G  
000652 004767 000000G  
000656 010067 000000G  
000662 000207 000000G  
000001

```
.SBTTL SETCTM - COUNTER TIMER
*** - SETCTM - SETUP COUNTER TIMER
INPUT:
 TEMP = TEMPLATE ADDRESS
 $OBUF = CURRENT POSITION IN OUTPUT BUFFER
OUTPUT:
-
SETCTM:
 MOV TEMP,R2 ; GET START OF TEMPLATE
 MOV C.CTM(R2),R1 ; GET NUMBER OF LOGICAL LINKS
 MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
 CALL $CNVD ; CONVERT NUMBER TO DECIMAL
 MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
 RETURN
.END
```

UPDLOG - CFE UPDATE LOGGING STA MACRO V05.03b Saturday 29-Jun-85 05:53 Page 8-1  
Symbol table

|                  |                       |                   |                   |                   |
|------------------|-----------------------|-------------------|-------------------|-------------------|
| CS.LOG= ***** GX | NEXT 000000R          | \$DIGIT= 000024   | \$MOVST= ***** GX | \$TMLST= ***** GX |
| CS.MOD= ***** GX | SETNUM 000122R        | \$DNUMB= 000014   | \$NUMBR= 000002   | \$UPLOG 000002RG  |
| C.LEN= 000005    | \$ALPHA= 000022       | \$EOS = 000012    | \$OBUF = ***** GX | \$\$\$FLG= 177777 |
| C.STA 000004     | \$ANY = 000020        | \$ERROR= ***** GX | \$OFDB = ***** GX | \$\$\$KEY= 177777 |
| C.STS = ***** GX | \$BLANK= 000006       | \$EXIT = 000000   | \$RAD50= 000016   | \$\$\$STA= 000000 |
| F.NRBD= ***** GX | \$BUFI = ***** GX     | \$FAIL = 177777   | \$STRNG= 000004   | .CONER= ***** GX  |
| LOGKTB 000000RG  | 003 \$CLNUP= ***** GX | \$LAMDA= 000000   | \$SUBXP= 000010   | .TPARS= ***** GX  |
| LOGSTB 000000RG  | 002 \$CNVO = ***** GX |                   |                   |                   |

. ABS. 000005 000 (RW,I,GBL,ABS,OVR)  
000152 001 (RW,I,LCL,REL,CON)  
\$STATE 000012 002 (RW,D,LCL,REL,CON)  
\$KTAB 000000 003 (RW,D,LCL,REL,CON)  
\$KSTR 000000 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10630 Words ( 42 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:09.21  
SY:UPDLOG.V2,[132,134]UPDLOG/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDLOG



```

296 .SBTTL SETHOA - SET HOST NODE AREA
297 ;+
298 *** - SETHOA - SET HOST NODE AREA
299
300 INPUT:
301 TEMP = TEMPLATE ADDRESS
302 (TEMP)+C.HOS = NODE ADDRESS
303
304 OUTPUT:
305 -
306
307
308 SETHOA:
309 000314 016700 000000G MOV $0BUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
310 000320 016702 177454 MOV TEMP,R2 ; POINT R2 AT TEMPLATE
311 000324 032762 000000G 000000G BIT #NO$HOA,C.STS(R2) ; HOST AREA PARSED ?
312 000332 001413 BEQ 10$; NOPE
313 000334 016201 000054 MOV C.HOS(R2),R1 ; GET HOST NODE ADDRESS
314 000340 042701 001777 BIC #1777,R1 ; ONLY AREA
315 000344 000301 SWAB R1 ; FORMAT IT
316 000346 006201 ASR R1 ; ...
317 000350 006201 ASR R1 ; ...
318 000352 004767 000000G CALL $CNVD ; CONVERT BINARY TO DECIMAL
319 000356 010067 000000G MOV R0,$0BUF ; UPDATE POINTER INTO OUTPUT BUFFER
320 000362 000207 10$: RETURN
321
322 .SBTTL SETHOS - SET HOST NODE ADDRESS
323 ;+
324 *** - SETHOS - SET HOST NODE ADDRESS
325
326 INPUT:
327 TEMP = TEMPLATE ADDRESS
328 (TEMP)+C.HOS = NODE ADDRESS
329
330 OUTPUT:
331 -
332
333
334 SETHOS:
335 000364 016700 000000G MOV $0BUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
336 000370 016702 177404 MOV TEMP,R2 ; POINT R2 AT TEMPLATE
337 000374 016201 000054 MOV C.HOS(R2),R1 ; GET HOST NODE ADDRESS
338 000400 042701 176000 BIC #C1777,R1 ; CLEAR AREA
339 000404 004767 000000G CALL $CNVD ; CONVERT BINARY TO DECIMAL
340 000410 010067 000000G MOV R0,$0BUF ; UPDATE POINTER INTO OUTPUT BUFFER
341 000414 000207 RETURN
342
343 .END
344
345 000001

```

```

272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287 000500
288 000500 016700 000000G
289 000504 016702 177270
290 000510 005001
291 000512 156201 000005
292 000516 042701 000007
293 000522 004767 000000G
294 000526 010067 000000G
295 000532 000207

 .SBTTL SETFLG - WRITE OUT THE FLAGS BYTE
 +
 *** - SETFLG - SET FLAGS BYTE
 :
 INPUT:
 :PNUMB = FLAGS BYTE
 :PNUMH = FLAGS BYTE (HIGH ORDER)
 :$OBUF = CURRENT POSITION IN OUTPUT LINE
 :TEMP = TEMPLATE ADDRESS
 :
 OUTPUT:
 :THE FLAGS BYTE IS WRITTEN TO THE OUTPUT LINE
 :-

SETFLG: MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
 MOV TEMP,R2 ; POINT R2 AT TEMPLATE
 CLR R1 ; GET THE FLAGS BYTE
 BISB C,FLG(R2),R1 ; ..WITHOUT SIGN EXTEND
 BIC #7,R1 ; GET THE FLAGS ONLY
 CALL $CNVO ; CONVERT NUMBER TO ASCII
 MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
 RETURN

```

186  
 187  
 188  
 189  
 190  
 191  
 192  
 193  
 194  
 195  
 196  
 197  
 198  
 199  
 200  
 201  
 202  
 203  
 204  
 205  
 206  
 207  
 208  
 209  
 210  
 211  
 212

000106  
 000106 016700 000000G  
 000112 016702 177662  
 000116 005762 000004  
 000122 001417  
 000124 016201 000004  
 000130 004767 000000G  
 000134 016702 177640  
 000140 005762 000006  
 000144 001406  
 000146 016702 177626  
 000152 016201 000006  
 000156 004767 000000G  
 000162 010067 000000G  
 000166 000207

```
.SBTTL SETNMO - SET UP PARTITION NAME 0
*** - SETNMO - SETUP PARTITION NAME 0
INPUT:
 (TEMP) = TEMPLATE ADDRESS
 (TEMP)+C.NAMO = RAD50 PARTITION NAME
OUTPUT:
-
SETNMO:
 MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
 MOV TEMP,R2 ; POINT AT TEMPLATE
 TST C.NAMO(R2) ; IS THERE A PARTITION NAME ?
 BEQ 10$; IF EQ, NO .. LEAVE
 MOV C.NAMO(R2),R1 ; GET RAD50 PARTITION NAME
 CALL $CSTA ; CONVERT RAD50 NAME TO ASCII
 MOV TEMP,R2 ; POINT AT TEMPLATE
 TST C.NAMO+2(R2) ; IS 2ND HALF OF NAME BLANK ?
 BEQ 10$; IF EQ, YES .. LEAVE
 MOV TEMP,R2 ; POINT AT TEMPLATE
 MOV C.NAMO+2(R2),R1 ; GET 2ND HALF OF PARTITION NAME
 CALL $CSTA ; CONVERT RAD50 NAME TO ASCII
 10$: MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
 RETURN
```

```

LOCAL DATA
122
123
124
125
126
127 000000
128 000000 000000
129 000000
130 000002
131 000004
132 000010
133 000012
134 000014
135 000016
136 000020
137 000000
138
139 000000 011 120 123 PSNDF: .ASCII <11>/PSN$DF/<11>
 000003 116 044 104
 000006 106 011
140 000010
141
142
143
144 000010
145 000010 000072' 000000G
146 000014 000000G 000000G
147 000020 000000G 000000G
148 000024 000000G 000000G
149 000030 000000G 000000G
150 000034 000000G 000000G
151 000040 000000G 000000G
152 000044 000000G 000000G
153 000050 000000G 000000G
154 000054 000000G 000000G
155 000060 000000G 000000G
156
157 000064 000000G 000000G
158 000070 000000
159

.SBTTL LOCAL DATA
::: DEFINE THE PSN BLOCK
.ASECT
.=0
.BLKW 1 ; C.LNK
.BLKW 1 ; C.STS
.C.NAM: .BLKW 2 ; RAD50 NETWORK NAME
.C.POR: .BLKW 1 ; NUMBER OF PORTS SUPPORTED (1-256)
.C.FLG: .BLKW 1 ; FLAGS
.C.TLO: .BLKW 1 ; TRANSPORT SUBADDRESS LOW
.C.THI: .BLKW 1 ; TRANSPORT SUBADDRESS HIGH
.C.LEN=. ; LENGTH OF THE BLOCK
.PSECT
PSNLF: .ASCII <11>/PSN$DF/<11>
PSNLF: .EVEN
PSNLF: .LIST BEX
DISTBL:
.WORD $UPPSN, CS.PSN
.WORD $UPX3P, CS.X3P
.WORD $UPDTE, CS.DTE
.WORD $UPCHN, CS.CHN
.WORD $UPPVC, CS.PVC
.WORD $UPCUG, CS.CUG
.WORD $UPDST, CS.DST
.WORD $UPDSA, CS.DSA
.WORD $UPDSC, CS.DSC
.WORD $UPRDT, CS.RDT
.WORD $UPX2P, CS.X2P
::: .WORD $UPSVC, CS.SVC
.WORD $UPX29, CS.X29
.WORD 0

```

UPDDDM CREATED BY MACRO ON 29-JUN-85 AT 05:49 PAGE 2 M 1  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL    | VALUE       | REFERENCES |
|-----------|-------------|------------|
| \$CAT5    | = ***** GX  | 7-248      |
| \$CNVD    | = ***** GX  | 10-328     |
| \$CNVO    | = ***** GX  | 9-306      |
| \$DIGIT   | = 000024    | #5-120     |
| \$DNUMB   | = 000014    | #5-120     |
| \$EOS     | = 000012    | #5-120     |
| \$ERROR   | = ***** GX  | *6-223     |
| \$EXIT    | = 000000    | #5-120     |
| \$FAIL    | = 177777    | #5-120     |
| \$GPRM    | = *****     | 5-120      |
| \$LAMDA   | = 000000    | #5-120     |
| \$MOVPT   | = ***** GX  | 7-245      |
| \$MOVST   | = ***** GX  | 6-216      |
| \$MVASC   | = ***** GX  | 8-273      |
|           |             | 8-282      |
| \$NUMBR   | = 000002    | #5-120     |
| \$OBUF    | = ***** GX  | 8-270      |
| \$OFDB    | = ***** GX  | *6-214     |
| \$RAD50   | = 000016    | #5-120     |
| \$RONLY   | = *****     | 5-120      |
| \$STRNG   | = 000004    | #5-120     |
| \$SUBXP   | = 000010    | #5-120     |
| \$TMLST   | = ***** GX  | 6-201      |
| \$UPDDM   | = 000102 RG | #6-200     |
| \$ZCOU    | = ***** GX  | 8-277      |
| \$ZCOUL   | = ***** GX  | 8-277      |
| \$ZMFL    | = ***** GX  | 8-275      |
| \$ZMFL    | = ***** GX  | 8-275      |
| \$ZTIM    | = ***** GX  | 8-274      |
| \$ZTIML   | = ***** GX  | 8-274      |
| \$\$\$FLG | = 177777    | #5-120     |
| \$\$\$KEY | = 177777    | #5-120     |
| .CONER    | = ***** GX  | 6-229      |
| .PSTPT    | = ***** GX  | 7-246      |
| .TPARS    | = ***** GX  | 6-221      |

SETINA - INACTIVITY TIMER

```

264 .SBTTL SETINA - INACTIVITY TIMER
265
266 :+
267 : *** - SETINA - SETUP INACTIVITY TIMER
268 :
269 : INPUT:
270 : TEMP = TEMPLATE ADDRESS
271 : $OBUF = CURRENT POSITION IN OUTPUT BUFFER
272 :
273 : OUTPUT:
274 :
275 :-
276
277 000250 016702 177524 SETINA: MOV TEMP,R2 ; GET START OF TEMPLATE
278 000254 005001 CLR R1 ; GET INACTIVITY TIMER
279 000256 156201 000014 BISB C.INAC(R2),R1 ;
280 000262 016700 000000G MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
281 000266 004767 000000G CALL $CNVD ; CONVERT NUMBER TO DECIMAL
282 000272 010067 000000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
283 000276 000207 RETURN

```

SETDEF - DELAY FACTOR

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52

.TITLE UPDDL - CFE UPDATE DLC PROCESS DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1982, 1983, 1984, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

#### MODULE DESCRIPTION:

STATE TABLE TO PARSE THE DLC DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

#### IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/RSX V1.0

UPDDL CREATED BY MACRO ON 29-JUN-85 AT 05:50 PAGE 3 M 4

MACRO CROSS REFERENCE CREF 04.00

| MACRO NAME | REFERENCES |        |        |        |        |        |        |        |        |        |
|------------|------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DBGTP\$    | #5-100     | #5-116 | #5-123 | #5-124 | #5-125 | #5-135 | #5-139 | #5-143 |        |        |
| ISTAT\$    | #4-67      | 5-100  |        |        |        |        |        |        |        |        |
| MTRANS     | #5-100     |        |        |        |        |        |        |        |        |        |
| PRINT      | #4-58      |        |        |        |        |        |        |        |        |        |
| STAT\$     | #4-67      | 5-102  | #5-105 | #5-108 | #5-111 | #5-114 | #5-118 | #5-121 | #5-127 | #5-130 |
|            | #5-133     | #5-137 | #5-141 | #5-145 | #5-148 | #5-151 | #5-156 | #5-159 |        |        |
| TRANS      | #4-67      | #5-103 | #5-106 | #5-109 | #5-112 | #5-115 | #5-116 | #5-119 | #5-122 | #5-123 |
|            | #5-124     | #5-125 | #5-128 | #5-131 | #5-134 | #5-135 | #5-138 | #5-139 | #5-142 | #5-143 |
|            | #5-146     | #5-149 | #5-152 | #5-157 |        |        |        |        |        |        |



.TITLE UPDDSC - UPDATE DESTINATION CALL PARAMETERS  
 .IDENT /V05.00/  
 .ENABL LC

COPYRIGHT (C) 1981, 1982, 1983, 1985 BY  
 DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
 ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
 INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
 COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
 OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
 TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
 AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
 CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
 SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

UPDATE DESTINATION CALL PARAMETERS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-AUG-81  
 DECNET-11M/S V3.1  
 DECNET-11M-PLUS V1.1
- 3.00 16-APR-82  
 DECNET-11M V3.1  
 DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
 DECNET-11M V4.0  
 DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
 DECnet-11M/S V4.2  
 DECnet-11M-Plus V3.0  
 DECnet-Micro/RSX V1.0

```

LOCAL DATA
131
132
133
134
135
136 000000
137 000000
138 000000
139 000002
140 000004
141 000006
142 000012
143 000013
144 000014
145 000020
146 000000
147
148 000000 011 104 123 DSTDF:
149 000010
150
151
152

.SBTTL LOCAL DATA
;
; DEFINE THE DESTINATION DESCRIPTOR BLOCK
;
.ASECT
.=0
.BLKW 1 ; C.LNK
.BLKW 1 ; C.STS
C.TYP: .BLKW 1 ; RAD50 'X25' OR 'X29'
C.NAM: .BLKW 2 ; RAD50 DESTINATION NAME
C.PRI: .BLKB 1 ; LIST INSERTION PRIORITY
C.OBJ: .BLKB 1 ; OBJECT NUMBER
C.TSK: .BLKW 2 ; RAD50 TASK NAME
C.LEN= ; LENGTH OF THE BLOCK

.PSECT
.NLIST BEX
.ASCII <11>/DST$DF/<11>
DSTLN=-DSTDF
.LIST BEX
.EVEN

```

UPDDTE - UPDATE LOCAL DTE DESCR MACRO V05.03b Saturday 29-Jun-85 05:51 M 7 Page 7-1  
Symbol table

|                  |                |                   |                   |                   |
|------------------|----------------|-------------------|-------------------|-------------------|
| CS.NET= ***** GX | C.LEN = 000040 | F.NRBD= ***** GX  | \$OBUF = ***** GX | \$WDATA= ***** GX |
| CS.TRB= ***** GX | C.LIN 000016   | ONSTR 000010R     | \$OFDB = ***** GX | .BUFF = ***** GX  |
| CS.UNI= ***** GX | C.NET 000034   | SETLIN 000260R    | \$PTNUM= ***** GX | .COMMA= ***** GX  |
| C.ADR 000004     | C.STA 000026   | SETSTA 000214R    | \$PTOPT= ***** GX | .FLAGS= ***** GX  |
| C.CT 000032      | DTEDF 000000R  | \$BUFI = ***** GX | \$PTR50= ***** GX | .TEMP = ***** GX  |
| C.HSH 000030     | DTELN = 000010 | \$MVASC= ***** GX | \$UPDTE 000014RG  |                   |

. ABS. 000040 000 (RW,I,GBL,ABS,OVR)  
000434 001 (RW,I,LCL,REL,CON)

Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 8846 Words ( 35 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:07.26

SY:UPDDTE.V2,[132,134]UPDDTE/CR/~SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDDTE

```

264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279 000440
280 000440 016702 177334
281 000444 016246 000016
282 000450 016700 000000G
283 000454 012701 000040
284 000460 012702 000007
285 000464 032716 000004
286 000470 001013
287 000472 012701 000047'
288 000476 012702 000004
289 000502 032716 000002
290 000506 001004
291 000510 012701 000053'
292 000514 012702 000007
293 000520 004767 000000G
294 000524 010067 000000G
295 000530 005726
296 000532 000207
297 000001

```

```

.SBTTL SETSNK - WRITE OUT SINK NAME
+
*** - SETSNK - WRITE OUT SINK NAME
:
INPUT:
: .PSTPT = START ADDRESS OF THE SINK NAME IN ASCII
: .PSTCN = NUMBER OF CHARACTERS IN THE SINK NAME
: $OBUF = CURRENT POSITION IN OUTPUT BUFFER
: TEMP = TEMPLATE ADDRESS
:
OUTPUT:
: THE SINK NAME IS WRITTEN INTO THE OUTPUT BUFFER
:
-
:
SETSNK:
MOV TEMP,R2 ; GET ADDRESS OF EVENT TEMPLATE
MOV C.SNK(R2),-(SP) ; GET SINK TYPE OF TEMPLATE
MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN LINE
MOV #MONITR,R1 ; POINT AT START OF MONITOR STRING
MOV #MONL,R2 ; GET LENGTH OF MONITOR STRING
BIT #FF.MON,(SP) ; MONITOR SINK TYPE?
BNE 10$; BR IF YES
MOV #FILE,R1 ; POINT AT START OF FILE STRING
MOV #FILEL,R2 ; GET LENGTH OF MONITOR STRING
BIT #FF.FIL,(SP) ; FILE SINK TYPE?
BNE 10$; BR IF YES
MOV #CONSOLE,R1 ; POINT AT START OF CONSOLE STRING
MOV #CONL,R2 ; GET LENGTH OF CONSOLE STRING
10$: CALL $MVASC ; MOVE INTO OUTPUT BUFFER
MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT LINE
TST (SP)+ ; CLEAN UP STACK
RETURN
.END

```

UPDFIL      CREATED BY MACRO ON 29-JUN-85 AT 05:52      PAGE 2      M 9  
MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|       |       |       |
|-------|-------|-------|
| RESRG | #4-48 | 6-166 |
| SAVRG | #4-48 | 6-156 |

.TITLE UPDLLC - CFE UPDATE LLC PROCESS DEFINITION IN CETAB  
.IDENT /V05.00/

COPYRIGHT (C) 1979, 1980, 1981, 1982, 1983, 1985 BY  
DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
TRANSFERRED.

THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
CORPORATION.

DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

MODULE DESCRIPTION:

STATE TABLE TO PARSE THE LLC DEFINITIONS

DISTRIBUTED SYSTEMS SOFTWARE ENGINEERING

IDENT HISTORY:

- 1.00 14-DEC-79  
DECNET-11M/S V3.0  
DECNET-11M-PLUS V1.0
- 3.00 16-APR-82  
DECNET-11M V3.1  
DECNET-11M-PLUS V1.1
- 4.00 07-NOV-83  
DECNET-11M V4.0  
DECNET-11M-PLUS V2.0
- 5.00 22-JUL-85  
DECnet-11M/S V4.2  
DECnet-11M-Plus V3.0  
DECnet-Micro/Rsx V1.0

Symbol table

|                  |                        |                   |                    |                      |
|------------------|------------------------|-------------------|--------------------|----------------------|
| BLTS 000016R     | 002 NLT 000130R        | 002 PRIOR 000070R | 002 ZLLCL = 000007 | \$MOVPT= ***** GX    |
| BITO 000030R     | 002 NM\$ARA= 176000    | SETCTM 000636R    | ZMTM 000011R       | \$MOVST= ***** GX    |
| CL.LEN= 000020   | NM\$NOD= 001777        | SETEXT 000610R    | ZMTML = 000007     | \$MVASC= ***** GX    |
| CS.LLC= ***** GX | NT\$AKD= 000020        | SETFLG 000224R    | ZSLI 000036R       | \$NUMBR= 000002      |
| CS.MOD= ***** GX | NT\$AKI= 000022        | SETLIN 000530R    | ZSLIL = 000007     | \$OBUF = ***** GX    |
| CTIM 000170R     | 002 NT\$CC = 000016    | SETNAM 000166R    | ZX3P 000027R       | \$OFDB = ***** GX    |
| C.CTM 000016     | NT\$CON= 000000        | SETNLT 000560R    | ZX3PL = 000007     | \$RAD50= 000016      |
| C.DEV 000012     | NT\$CTL= 000000        | SETPRI 000500R    | \$ALPHA= 000022    | \$STRNG= 000004      |
| C.EXT 000015     | NT\$DAT= 000002        | TEMP 000000R      | \$ANY = 000020     | \$SUBXP= 000010      |
| C.FLG 000006     | NT\$DC = 000012        | ZCOU 000020R      | \$BLANK= 000006    | \$TMLST= ***** GX    |
| C.LIN 000011     | NT\$DIS= 000014        | ZCOUL = 000007    | \$BUF1 = ***** GX  | \$UPLLC 000054RG     |
| C.NAM 000004     | NT\$DLS= 000006        | ZF 000206R        | \$CAT5 = ***** GX  | \$ZMFL = ***** GX    |
| C.NCTD= 000013   | NT\$ILS= 000010        | ZF.COUL= ***** GX | \$CLNUP= ***** GX  | \$ZMFL = ***** GX    |
| C.NLT 000014     | NT\$IMS= 000002        | ZF.INI= ***** GX  | \$CNVD = ***** GX  | \$ZTIM = ***** GX    |
| C.NLTD= 000012   | NT\$INT= 000004        | ZF.MFL= ***** GX  | \$CNVD = ***** GX  | \$ZTIML= ***** GX    |
| C.PRI 000010     | NT\$RET= 000032        | ZF.MTM= ***** GX  | \$DITG= 000024     | \$\$\$\$FLG= 177777  |
| C.STS = ***** GX | NT\$ROU= 000024        | ZF.SLI= ***** GX  | \$DNUMB= 000014    | \$\$\$\$KEY= 000010  |
| EXIT 000214R     | 002 NT\$RTR= 000030    | ZF.TIM= ***** GX  | \$EOS = 000012     | \$\$\$\$STA= 000000  |
| EXT 000150R      | 002 NT\$TSP= 000026    | ZF.X3P= ***** GX  | \$ERROR= ***** GX  | \$\$\$\$TMP= 000040R |
| F.NRBD= ***** GX | N\$\$\$ACK= 000011     | ZINI 000045R      | \$EXIT = 000000    | .CONER= ***** GX     |
| LLCKTB 000000RG  | 003 N\$\$\$HDR= 000007 | ZINIL = 000007    | \$FAIL = 177777    | .PSTPT= ***** GX     |
| LLCSTB 000000RG  | 002 N\$\$\$OVR= 000022 | ZLLC 000002R      | \$LAMDA= 000000    | .TPARS= ***** GX     |

004

. ABS. 000020 000 (RW,I,GBL,ABS,OVR)  
 000664 001 (RW,I,LCL,REL,CON)  
 \$STATE 000222 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000022 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000043 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

### \*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10760 Words ( 43 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:44.12  
 SY:UPDLLC.V2,[132,134]UPDLLC/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDLLC

UPDLOG      CREATED BY    MACRO    ON 29-JUN-85 AT 05:53      PAGE 1      M 12

SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE       | REFERENCES                                            |
|-----------|-------------|-------------------------------------------------------|
| CS.LOG    | = ***** GX  | 7-148                                                 |
| CS.MOD    | = ***** GX  | 7-155                                                 |
| C.LEN     | = 000005    | #5-106                                                |
| C.STA     | = 000004    | #5-105      7-159                                     |
| C.STS     | = ***** GX  | 7-148      7-155                                      |
| F.NRBD    | = ***** GX  | *7-157                                                |
| LOGKTB    | = 000000 RG | #6-116      7-163                                     |
| LOGSTB    | = 000000 RG | #6-116      7-164                                     |
| NEXT      | = 000000 R  | #5-109      *7-154      *7-159      8-191      *8-193 |
| SETNUM    | = 000122 R  | #8-189                                                |
| \$ALPHA   | = 000022    | #6-116                                                |
| \$ANY     | = 000020    | #6-116                                                |
| \$BLANK   | = 000006    | #6-116                                                |
| \$BUFI    | = ***** GX  | 7-157                                                 |
| \$CNVO    | = ***** GX  | 8-194                                                 |
| \$DIGIT   | = 000024    | #6-116                                                |
| \$DNUMB   | = 000014    | #6-116                                                |
| \$EOS     | = 000012    | #6-116                                                |
| \$ERROR   | = ***** GX  | *7-167                                                |
| \$EXIT    | = 000000    | #6-116                                                |
| \$FAIL    | = 177777    | #6-116                                                |
| \$GPRM    | = *****     | 6-116                                                 |
| \$LAMDA   | = 000000    | #6-116                                                |
| \$MOVST   | = ***** GX  | 7-160                                                 |
| \$NUMBR   | = 000002    | #6-116                                                |
| \$OBUF    | = ***** GX  | 8-190      *8-195                                     |
| \$OFDB    | = ***** GX  | *7-157                                                |
| \$RAD50   | = 000016    | #6-116                                                |
| \$RONLY   | = *****     | 6-116      6-116                                      |
| \$STRNG   | = 000004    | #6-116                                                |
| \$SUBXP   | = 000010    | #6-116                                                |
| \$TMLST   | = ***** GX  | 7-144                                                 |
| \$UPLOG   | = 000002 RG | #7-143                                                |
| \$\$\$FLG | = 177777    | #6-116                                                |
| \$\$\$KEY | = 177777    | #6-116                                                |
| .CONER    | = ***** GX  | 7-173                                                 |
| .TPARS    | = ***** GX  | 7-165                                                 |



Symbol table

|                  |                    |                     |                   |                    |
|------------------|--------------------|---------------------|-------------------|--------------------|
| ADD2 000052R     | 002 ID 000116R     | 002 SETNNA 000212R  | \$DNUMB= 000014   | \$QFDB = ***** GX  |
| CS.MOD= ***** GX | LA = 000074        | SETNNM 000262R      | \$EQS = 000012    | \$RAD50= 000016    |
| CS.NOD= ***** GX | NODKTB 000000RG    | 003 TEMP 000000R    | \$ERROR= ***** GX | \$STRNG= 000004    |
| C.HOS 000054     | NODSTB 000000RG    | 002 \$ALPHA= 000022 | \$EXIT = 000000   | \$SUBXP= 000010    |
| C.LEN = 000056   | NO\$EXA= ***** GX  | \$ANY = 000020      | \$FAIL = 177777   | \$TMLST= ***** GX  |
| C.NAM 000004     | NO\$HOA= ***** GX  | \$BLANK= 000006     | \$LAMD= 000000    | \$UPNOD 000002RG   |
| C.NID 000010     | RA = 000076        | \$BUFI = ***** GX   | \$MOVPT= ***** GX | \$\$\$FLG= 177777  |
| C.NUM 000052     | SETHOA 000314R     | \$CLNUP= ***** GX   | \$MOVST= ***** GX | \$\$\$KEY= 177777  |
| C.STS = ***** GX | SETHOS 000364R     | \$CNVD = ***** GX   | \$MVASC= ***** GX | \$\$\$STA= 000116R |
| EXIT 000110R     | 002 SETNAM 000114R | \$CSTA = ***** GX   | \$NUMBR= 000002   | .CONER= ***** GX   |
| F.NRBD= ***** GX | SETNID 000156R     | \$DIGIT= 000024     | \$OBUF = ***** GX | .TPARS= ***** GX   |
| HOS2 000100R     | 002                |                     |                   |                    |

. ABS. 000056 000 (RW,I,GBL,ABS,OVR)  
 000416 001 (RW,I,LCL,REL,CON)  
 \$STATE 000126 002 (RW,D,LCL,REL,CON)  
 \$KTAB 000000 003 (RW,D,LCL,REL,CON)  
 \$KSTR 000000 004 (RW,D,LCL,REL,CON)  
 Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
 Work file writes: 0  
 Size of work file: 10422 Words ( 41 Pages)  
 Size of core pool: 14440 Words ( 55 Pages)  
 Operating system: RSX-11M/PLUS

Elapsed time: 00:00:28.05  
 SY:UPDNOD.V2,[132,134]UPDNOD/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDNOD

002

```

297 .SBTTL SETCPY - WRITE OUT THE MAXIMUM NUMBER OF COPIES ALLOWED
298
299 ;+
300 *** - SETCPY - WRITE OUT THE MAXIMUM NUMBER OF COPIES ALLOWED
301
302 INPUT:
303 .PNUMB = THE MAX. # OF COPIES ALLOWED
304 .PNUMH = THE MAX. # OF COPIES ALLOWED (HIGH BYTE)
305 $OBUF = CURRENT POSITION IN OUTPUT LINE
306 TEMP = TEMPLATE ADDRESS
307
308 OUTPUT:
309 THE NUMBER OF COPIES IS WRITTEN TO THE OUTPUT BUFFER
310
311 -
312
313 000534 016700 000000G SETCPY: MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
314 000540 016702 177234 MOV TEMP,R2 ; POINT AT TEMPLATE
315 000544 116201 000012 MOV C.CPY(R2),R1 ; GET THE CURRENT NUMBER OF COPIES ALLOWED
316 000550 004767 000000G CALL $CNVD ; CONVERT NUMBER TO ASCII
317 000554 010067 000000G MOV R0,$OBUF ; UPDATE CURRENT POSITION IN OUTPUT BUFFER
318 000560 000207
319
320
321 000001 .END

```

```

214
215
216
217
218
219
220
221
222
223
224
225
226 000170
227 000170 016700 000000G
228 000174 016702 177600
229 000200 016201 000010
230 000204 004767 000000G
231 000210 005761 000012
232 000214 001406
233 000216 016702 177556
234 000222 016201 000012
235 000226 004767 000000G
236 000232 010067 000000G
237 000236 000207
238

.SBTTL SETNM1 - SET UP PARTITION NAME 1
*** - SETNM1 - SETUP PARTITION NAME 1
INPUT:
TEMP = TEMPLATE ADDRESS
(TEMP)+C.NAM1 = RAD50 PARTITION NAME
OUTPUT:
-
SETNM1:
MOV $OBUF,R0 ; POINT AT CURRENT POSITION IN OUTPUT BUFFER
MOV TEMP,R2 ; GET TEMPLATE ADDRESS
MOV C.NAM1(R2),R1 ; GET RAD50 PARTITION NAME
CALL $C5TA ; CONVERT RAD50 NAME TO ASCII
TST C.NAM1+2(R1) ; IS 2ND HALF OF NAME BLANK ?
BEQ 10$; IF EQ, YES, LEAVE
MOV TEMP,R2 ; POINT AT TEMPLATE
MOV C.NAM1+2(R2),R1 ; GET 2ND HALF OF PARTITION NAME
CALL $C5TA ; CONVERT RAD50 TO ASCII
10$: MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
RETURN

```

```

161 .SBTTL $UPPSN - UPDATE PSN DEFINITION
162
163 *** - $UPPSN - UPDATE PSN DEFINITION
164
165 INPUT:
166 R3-R5 - TPARS REGISTERS
167
168 OUTPUT:
169 C-BIT = SUCCESS/FAILURE
170 ALL PSI TEMPLATES UPDATED
171
172 ACTIONS
173 UPDATES PSN$DF FROM TEMPLATE. ALL PSI TEMPLATES ARE
174 UPDATED FROM THE INTERNAL LIST.
175
176 -
177
178 $UPPSN:
179 000072 016700 000000G MOV $PSIPT,R0 ; POINT TO PSN$DF TEMPLATE
180 000076 001533 BEQ 101$; IF EQ, INTERNAL PROBLEM
181 000100 010067 000000G 10$: MOV R0,TEMP ; SAVE POINTER
182 000104 005720 TST (R0)+ ; POINT TO C.STS
183 000106 010067 000000G MOV R0,.FLAGS ; SAVE FOR OPTION TESTING
184 000112 012700 000000G MOV #.BUFF,R0 ; POINT AT START OF OUTPUT BUFFER
185 000116 010067 000000C MOV R0,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
186 000122 012701 000000' MOV #PSNDF,R1 ; POINT AT START OF X3P DEFINITION
187 000126 012702 000010 MOV #PSNLN,R2 ; GET LENGTH OF X3P DEFINITION
188 000132 004767 000000G CALL $MVASC ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
189 000136 010067 000000G MOV R0,$OBUF ; SAVE CURRENT POSITION IN OUTPUT BUFFER
190 000142 010546 MOV R5,-(SP) ; SAVE R5
191 000144 012767 177777 000000G MOV #-1,.COMMA ; SUPPRESS .INITIAL COMMA
192 000152 $PTR50 ; NAME
193 000166 $PTNUM ; #C.POR ; PORTS
194 000200 $PTNUM ; #C.FLG,OCTAL,#CS.FLG ; OPTIONAL FLAGS
195 000224 $PTNUM ; #C.TLO,,#CS.TLO ; OPTIONAL TRANSPORT LOW SUBADDRESS
196 000250 $PTNUM ; #C.THI,,#CS.THI ; OPTIONAL TRANSPORT HIGH SUBADDRESS
197
198 ; PROCESS REMAINING PSI TEMPLATES
199
200 15$: MOV $OBUF,R1 ; CALCULATE LENGTH OF BUFFER
201 SUB #.BUFF,R1 ;
202 MOV R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
203 CALL $WDATA ; WRITE THE RECORD TO THE OUTPUT FILE
204 20$: MOV @TEMP,R5 ; POINT TO PSN$DF TEMPLATE
205 BEQ 40$; IF EQ, NO MORE TEMPLATES
206 MOV #DISTBL,R0 ; POINT TO DISPATCH TABLE
207 MOVB C.STS(R5),R1 ; GET TEMPLATE TYPE CODE
208 30$: MOV (R0)+,R2 ; PICK UP ROUTINE ADDRESS
209 BEQ 35$; IF EQ, END OF PSI TEMPLATES
210 CMP R1,(R0)+ ; CHECK TYPE
211 BNE 30$; IF NE, NEXT ENTRY
212 CALL (R2) ; ELSE DISPATCH TO UPDATE ROUTINE
213 BR 20$; PROCESS NEXT TEMPLATE
214 35$: MOV R5,TEMP ; POINT TO NEXT TEMPLATE IN LIST
215 BR 20$; CONTINUE SCAN
216
217 40$: MOV (SP)+,R5 ; RESTORE R5
218 MOV #BUI1,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE 'END$DF' LINE

```

CREF 04.00

| MACRO NAME | REFERENCES                          |                            |                            |                            |                            |                            |                            |                            |                            |                            |
|------------|-------------------------------------|----------------------------|----------------------------|----------------------------|----------------------------|----------------------------|----------------------------|----------------------------|----------------------------|----------------------------|
| DBGTP\$    | #5-120<br>#5-151                    | #5-136<br>#5-152           | #5-143<br>#5-162           | #5-144<br>#5-166           | #5-145<br>#5-170           | #5-146                     | #5-147                     | #5-148                     | #5-149                     | #5-150                     |
| FLAG       | #4-64                               | 4-90                       | 4-91                       | 4-92                       | 4-93                       | 4-94                       | 4-95                       | 4-96                       | 4-97                       | 4-98                       |
| I\$TAT\$   | #4-82                               | 5-120                      |                            |                            |                            |                            |                            |                            |                            |                            |
| MTRANS     | #5-120                              |                            |                            |                            |                            |                            |                            |                            |                            |                            |
| PRINT      | #4-59                               |                            |                            |                            |                            |                            |                            |                            |                            |                            |
| POTFLG     | #4-69<br>8-283                      | 8-274<br>8-284             | 8-275                      | 8-276                      | 8-277                      | 8-278                      | 8-279                      | 8-280                      | 8-281                      | 8-282                      |
| STATES     | #4-82<br>#5-160                     | 5-122<br>#5-164            | #5-125<br>#5-168           | #5-128<br>#5-172           | #5-131<br>#5-175           | #5-134<br>#5-178           | #5-138<br>#5-183           | #5-141<br>#5-186           | #5-154                     | #5-157                     |
| TRANS      | #4-82<br>#5-144<br>#5-158<br>#5-184 | #5-123<br>#5-145<br>#5-161 | #5-126<br>#5-146<br>#5-162 | #5-129<br>#5-147<br>#5-165 | #5-132<br>#5-148<br>#5-166 | #5-135<br>#5-149<br>#5-169 | #5-136<br>#5-150<br>#5-170 | #5-139<br>#5-151<br>#5-173 | #5-142<br>#5-152<br>#5-176 | #5-143<br>#5-155<br>#5-179 |

```

285 .SBTTL SETDEF - DELAY FACTOR
286 ;+
287 ;*** - SETDEF - SETUP DELAY FACTOR
288 ;
289 ; INPUT:
290 ; TEMP = TEMPLATE ADDRESS
291 ; $OBUF = CURRENT POSITION IN OUTPUT BUFFER
292 ;
293 ; OUTPUT:
294 ;
295 ; -
296
297 000300 SETDEF:
298 000300 016702 177474 MOV TEMP,R2 ; GET START OF TEMPLATE
299 000304 005001 CLR R1 ; GET DELAY FACTOR
300 000306 156201 000016 BISB C,DELF(R2),R1 ;
301 000312 016700 000000G MOV $OBUF,R0 ; GET CURRENT POSITION INTO OUTPUT BUFFER
302 000316 0047 7 000000G CALL $CNVD ; CONVERT NUMBER TO DECIMAL
303 000322 01006 600000G MOV R0,$OBUF ; UPDATE POINTER INTO OUTPUT BUFFER
304 000326 000207 RETURN

```

```
54 ;****
55 ; LOCAL MACROS
56 ;****
57
58 .MACRO PRINT TEXT
59 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
60 CALL $PRINT ; PRINT MESSAGE
61 .ENDM PRINT
62
63 ;****
64 ; MACRO CALLS
65 ;****
66
67 .MCALL ISTAT$,STATE$,TRAN$
68
69 ;****
70 ; LOCAL DATA
71 ;****
72
73 000000 TEMP: .BLKW 1 ; ADDRESS OF DLC TEMPLATE
74 000002 132 106 056 ZDLC: .ASCII /ZF.DLC!/ ; MARK PROCESS AN A DLC
75 000005 104 114 103
76 000010 041
77 000007 ZMAN: ZDLC=-ZDLC ; AND LENGTH
78 000011 132 106 056 .ASCII /ZF.MAN!/ ; SUPPORT FOR NETWORK MANAGEMENT FUNCTIONS
79 000014 115 101 116
80 000017 041
81 000007 ZMANL=-ZMAN ; AND LENGTH
82 .EVEN
83
84 ;
85 ; DEFINE DLC TEMPLATE
86 ;
87 000000 .ASECT
88 000000 .=0
89 000002 .BLKW 1 ; C.LNK
90 000004 .BLKW 1 ; C.STS
91 000006 C.NAM: .BLKW 1 ; DLC PROCESS NAME
92 000010 C.FLG: .BLKW 1 ; DLC FLAGS WORD
93 000012 C.PRI: .BLKW 1 ; DLC PROCESS PRIORITY
94 000013 C.NLT: .BLKB 1 ; NUMBER OF LINE TABLES IN PROCESS SPACE
95 000014 C.NCT: .BLKB 1 ; NUMBER OF CONTROLLER TABLES
96 000020 C.LEN=. .PSECT
```

\*\*FILE\*\*ID\*\*UPDDSA

|            |            |         |          |          |          |           |      |
|------------|------------|---------|----------|----------|----------|-----------|------|
| UU         | UU         | PPPPPPP | DDDDDDDD | DDDDDDDD | SSSSSSSS | AAAAAA    |      |
| UU         | UU         | PPPPPPP | DDDDDDDD | DDDDDDDD | SSSSSSSS | AAAAAA    |      |
| UU         | UU         | PP PP   | DD DD    | DD DD    | SS       | AA AA     |      |
| UU         | UU         | PP PP   | DD DD    | DD DD    | SS       | AA AA     |      |
| UU         | UU         | PP PP   | DD DD    | DD DD    | SS       | AA AA     |      |
| UU         | UU         | PP PP   | DD DD    | DD DD    | SS       | AA AA     |      |
| UU         | UU         | PPPPPPP | DD DD    | DD DD    | SSSSSS   | AA AA     |      |
| UU         | UU         | PPPPPPP | DD DD    | DD DD    | SSSSSS   | AA AA     |      |
| UU         | UU         | PP      | DD DD    | DD DD    | SS       | AAAAAAAAA |      |
| UU         | UU         | PP      | DD DD    | DD DD    | SS       | AAAAAAAAA |      |
| UU         | UU         | PP      | DD DD    | DD DD    | SS       | AA AA     | .... |
| UU         | UU         | PP      | DD DD    | DD DD    | SS       | AA AA     | .... |
| UUUUUUUUUU | UUUUUUUUUU | PP      | DDDDDDDD | DDDDDDDD | SSSSSSSS | AA AA     | .... |
| UUUUUUUUUU | UUUUUUUUUU | PP      | DDDDDDDD | DDDDDDDD | SSSSSSSS | AA AA     | .... |

|            |          |           |
|------------|----------|-----------|
| LL         | SSSSSSSS | TTTTTTTTT |
| LL         | SSSSSSSS | TTTTTTTTT |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SSSSSS   | TT        |
| LL         | SSSSSS   | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LL         | SS       | TT        |
| LLLLLLLLLL | SSSSSSSS | TT        |
| LLLLLLLLLL | SSSSSSSS | TT        |



```

57 .SBTTL MACRO CALLS AND LOCAL DATA
58 .SBTTL MACRO DEFINITIONS
59
60 ; LOCAL MACROS
61
62 .MACRO PRINT TEXT
63 MOV TEXT,R0 ; POINT R0 AT ADDRESS OF MESSAGE TO PRINT
64 CALL $PRINT ; PRINT MESSAGE
65 .ENDM PRINT
66
67 .MACRO ERROR$ TEXT
68
69 ; THE ERROR TEXT MUST HAVE THE ERROR STATUS AS THE FIRST BYTE OF THE ERROR
70 ; MESSAGE STRING.
71
72 .IF DIF <TEXT><R0>
73 MOV TEXT,R0 ; GET ADDRESS OF ERROR MESSAGE
74 .ENDC
75 CALL $CFERR ; PRINT ERROR MESSAGE
76 .ENDM ERROR$
77
78 .MACRO EPRINT TEXT
79
80 ; PRINT TEXT ON ERROR LUN
81
82 .IF DIF <TEXT><R0>
83 MOV TEXT,R0 ; GET ADDRESS OF ASCIZ STRING TO PRINT
84 .ENDC
85 CALL $CFER1 ; PRINT MESSAGE ON ERROR LUN
86 .ENDM EPRINT
87
88 ; PUT NUMERIC FROM TEMPLATE TO OUTPUT BUFFER
89
90 .MACRO $$$PTN ROU
91 .MACRO 'ROU OFS,OCTL,OPT,?L
92 .IF NB <OPT>
93 MOV OPT,R2
94 CALL $PTOPT
95 BCS L
96 .IFTF
97 MOV OFS,R2
98 .IF NB <OCTL>
99 MOV SP,R1
100 .IFF
101 CLR R1
102 .ENDC ; IF NB <OCTL>
103 CALL 'ROU
104 .IFT ; IF NB <OPT>
105 L:
106 .ENDC ; IF NB <OPT>
107 .ENDM ; 'ROU
108 .ENDM ; $$$PTN
109
110 $$$PTN $PTNUM
111 $$$PTN $PTNMB
112
113 ;

```

```

154 .SBTTL $UPDST - UPDATE DESTINATION DESCRIPTOR
155 +
156 *** - $UPDST - UPDATE DESTINATION DESCRIPTOR
157 :
158 INPUT:
159 R3-R5 - TPARS REGISTERS
160 :
161 OUTPUT:
162 C-BIT = SUCCESS/FAILURE
163 :
164 -
165
166 000010 $UPDST::
167 000010 012767 000000G 000000C MOV #.BUFF,$OFDB+F.NRBD+2 ; SET NEW OUTPUT BUFFER
168 000016 010567 000000G MOV R5,TEMP ; SAVE TEMPLATE ADDRESS
169 000022 005725 TST (R5)+ ; POINT TO C.STS
170 000024 010567 000000G MOV R5,.FLAGS ; SAVE ADDRESS OF FLAGS WORD
171 000030 012767 177777 000000G MOV #-1,.COMMA ; INITIALIZE FOR OPTIONS
172 000036 012700 000000G MOV #.BUFF,R0 ; POINT AT START OF OUTPUT BUFFER
173 000042 012701 000000G MOV #DSTDF,R1 ; POINT AT START OF DST DEFINITION
174 000046 012702 000010 MOV #DSTLN,R2 ; GET LENGTH OF DST DEFINITION
175 000052 004767 000000G CALL $MVASC ; MOVE 1ST PART OF LINE INTO OUTPUT BUFFER
176 000056 010067 000000G MOV R0,$OBUF ; SAVE CURRENT POSITION IN OUTPUT BUFFER
177 000062 $PTR50 #C.TYP,#1 ; DESTINATION TYPE
178 000076 $PTR50 #C.NAM,#2 ; DESTINATION NAME
179 000112 $PTNMB #C.PRI ; LIST INSERTION PRIORITY
180 000124 $PTNMB #C.OBJ ; OBJECT NUMBER
181 000136 $PTR50 #C.TSK,#2,#CS.TSK ; TASK NAME
182 000164 016701 000000G MOV $OBUF,R1 ; CALCULATE LENGTH OF BUFFER
183 000170 162701 000000G SUB #.BUFF,R1 ;
184 000174 010167 000000C MOV R1,$OFDB+F.NRBD ; SET LENGTH OF BUFFER
185 000200 004767 000000G CALL $WDATA ; WRITE THE RECORD TO THE OUTPUT FILE
186 000204 012767 000000G 000000C MOV #$.BUF I,$OFDB+F.NRBD+2 ; SET TO OUTPUT THE "END$DF" LINE
187 000212 000207 RETURN
188
189
190 000001 .END

```

UPDDTE      CREATED BY    MACRO    ON 29-JUN-85 AT 05:51      PAGE 1      N 7  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL  | VALUE      | REFERENCES                                          |
|---------|------------|-----------------------------------------------------|
| CS.NET  | = ***** GX | 6-193                                               |
| CS.TRB  | = ***** GX | 7-237                                               |
| CS.UNI  | = ***** GX | 7-231                                               |
| C.ADR   | 000004     | #5-144 6-188                                        |
| C.CT    | 000032     | #5-149 6-191                                        |
| C.HSH   | 000030     | #5-148 6-190                                        |
| C.LEN   | = 000040   | #5-151                                              |
| C.LIN   | 000016     | #5-145 7-226 7-230 7-236 7-242                      |
| C.NET   | 000034     | #5-150 6-193                                        |
| C.STA   | 000026     | #5-147 7-217                                        |
| DIEDF   | 000000 R   | #5-155 5-156 6-184                                  |
| DTELN   | = 000010   | #5-156 6-185                                        |
| F.NRBD  | = ***** GX | *6-178 *6-196 *6-198                                |
| ONSTR   | 000010 R   | #5-158 7-220                                        |
| SETLIN  | 000260 R   | 6-189 #7-226                                        |
| SETSTA  | 000214 R   | 6-192 #7-214                                        |
| \$BUFI  | = ***** GX | 6-198                                               |
| \$MVASC | = ***** GX | 6-186                                               |
| \$OBUF  | = ***** GX | *6-187 6-194 7-227 *7-228 7-233 *7-234 7-239 *7-240 |
| \$OFDB  | = ***** GX | *6-178 *6-196 *6-198                                |
| \$PTNUM | = ***** GX | 6-190 6-191 7-230 7-236 7-242                       |
| \$PTOPT | = ***** GX | 6-193                                               |
| \$PTRSO | = ***** GX | 6-188 6-193 7-222 7-226                             |
| \$UPDTE | 000014 RG  | #6-177                                              |
| \$WDATA | = ***** GX | 6-197                                               |
| .BUFF   | = ***** GX | 6-178 6-183 6-195                                   |
| .COMMA  | = ***** GX | *6-182 *7-229 *7-235 *7-241                         |
| .FLAGS  | = ***** GX | *6-181 7-231 7-237                                  |
| .TEMP   | = ***** GX | *6-179 7-214 *7-220 *7-223                          |

|                  |                    |                  |                   |                    |
|------------------|--------------------|------------------|-------------------|--------------------|
| BUFF 000062R     | EXIT 000032R       | 002 F.CEV 000030 | SETCLS 000360R    | \$LAMDA= 000000    |
| CONL = 000007    | FF.ADD= 000020     | F.CIR 000024     | SETMSK 000400R    | \$MOVPT= ***** GX  |
| CONSLE 000053R   | FF.CIR= 000040     | F.CLS 000002     | SETNUM 000422R    | \$M'ASC= ***** GX  |
| CS.EVT= ***** GX | FF.CON= 000001     | F.EVT 000004     | SETSNK 000440R    | \$NUMBER= 000002   |
| C.CLS 000004     | FF.FIL= 000002     | F.FLG 000014     | TEMP 000000R      | \$OBUF = ***** GX  |
| C.EVT 000006     | FF.HST= 040000     | F.LEN 000016     | \$ALPHA= 000022   | \$OFDB = ***** GX  |
| C.LEN = 000020   | FF.LIN= 000010     | F.LIN 000020     | \$ANY = 000020    | \$RAD50= 000016    |
| C.SNK 000016     | FF.MOD= 000100     | F.LNK 000000     | \$BLANK= 000006   | \$STRNG= 000004    |
| C.STS = ***** GX | FF.MON= 000004     | F.MOD 000022     | \$BUFI = ***** GX | \$SUBXP= 000010    |
| EVTDEF 000004R   | FF.MSK= 000077     | F.NRBD= ***** GX | \$CNVO = ***** GX | \$TMLST= ***** GX  |
| EVTDF 000030R    | FF.PRT= 000200     | F.REM 000026     | \$DIGIT= 000024   | \$UPEVT 000202RG   |
| EVTKITB 000000RG | 003 FF.QUL= 000370 | F.SEV 000004     | \$DNUMB= 000014   | \$WDATA= ***** GX  |
| EVTLEN= 000023   | FF.REM= 100000     | LA = 000074      | \$EOS = 000012    | \$\$\$FLG= 177777  |
| EVTLN = 000010   | FILE 000047R       | MONITR 000040R   | \$ERROR= ***** GX | \$\$\$KEY= 177777  |
| EVTMSK 000036R   | 002 FILEL = 000004 | MONL = 000007    | \$EXIT = 000000   | \$\$\$STA= 000036R |
| EVTOFF 000002R   | F.ADD 000016       | RA = 000076      | \$FAIL = 177777   | .TPARS= ***** GX   |
| EVTSTB 000000RG  | 002                |                  |                   |                    |

. ABS. 000040 000 (RW,I,GBL,ABS,OVR)  
000534 001 (RW,I,LCL,REL,CON)  
\$STATE 000056 002 (RW,D,LCL,REL,CON)  
\$KTAB 000000 003 (RW,D,LCL,REL,CON)  
\$KSTR 000000 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

### \*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 11216 Words ( 44 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:19.30  
SY:UPDEVT.V2,[132,134]UPDEVT/CR/-SP=SY:[1,1]RSXMCM.SML/ML,[130,110]NETLIB/ML,[132,10]UPDEVT

\*\*FILE\*\*ID\*\*UPDFA

|           |    |         |          |           |            |            |        |
|-----------|----|---------|----------|-----------|------------|------------|--------|
| UU        | UU | PPPPPPP | DDDDDDDD | FFFFFFFFF | EEEEEEEEEE | AAAAAA     |        |
| UU        | UU | PPPPPPP | DDDDDDDD | FFFFFFFFF | EEEEEEEEEE | AAAAAA     |        |
| UU        | UU | PP      | DD       | DD        | FF         | AA         | AA     |
| UU        | UU | PP      | DD       | DD        | FF         | AA         | AA     |
| UU        | UU | PP      | DD       | DD        | FF         | AA         | AA     |
| UU        | UU | PP      | DD       | DD        | FF         | AA         | AA     |
| UU        | UU | PPPPPPP | DD       | DD        | FFFFFFFFF  | EEEEEEEEEE | AAAAAA |
| UU        | UU | PPPPPPP | DD       | DD        | FFFFFFFFF  | EEEEEEEEEE | AAAAAA |
| UU        | UU | PP      | DD       | DD        | FF         | AA         | AA     |
| UU        | UU | PP      | DD       | DD        | FF         | AAAAAAAAAA |        |
| UU        | UU | PP      | DD       | DD        | FF         | AAAAAAAAAA |        |
| UU        | UU | PP      | DD       | DD        | FF         | AA         | AA     |
| UU        | UU | PP      | DD       | DD        | FF         | AA         | AA     |
| UU        | UU | PP      | DD       | DD        | FF         | AA         | AA     |
| UUUUUUUUU | UU | PP      | DDDDDDDD | FF        | EEEEEEEEEE | AA         | AA     |
| UUUUUUUUU | UU | PP      | DDDDDDDD | FF        | EEEEEEEEEE | AA         | AA     |

|           |          |           |
|-----------|----------|-----------|
| LL        | SSSSSSSS | TTTTTTTTT |
| LL        | SSSSSSSS | TTTTTTTTT |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SSSSSS   | TT        |
| LL        | SSSSSS   | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LL        | SS       | TT        |
| LLLLLLLLL | SSSSSSS  | TT        |
| LLLLLLLLL | SSSSSSS  | TT        |

```

53 ;****
54 ; LOCAL MACROS
55 ;****
56
57 .MACRO PRINT TEXT
58 MOV TEXT,RO ; POINT RO AT ADDRESS OF MESSAGE TO PRINT
59 CALL $PRINT ; PRINT MESSAGE
60 .ENDM PRINT
61
62 ;****
63 ; MACRO CALLS
64 ;****
65
66 .MCALL ISTAT$,STATE$,TRAN$,MSGDF$,LLCTP$
67
68 000000 LLCTP$ LIST ; DEFINE LLC$DF TEMPLATE OFFSETS
 ;
 ; DEFINE LLC TEMPLATE
 ;
 .ASECT
 .=0
 .BLKW 1 ; C.LNK
 .BLKW 1 ; C.STS
 C.NAM: .BLKW 1 ; LLC PROCESS NAME
 C.FLG: .BLKW 1 ; LLC FLAGS WORD
 C.PRI: .BLKW 1 ; LLC PROCESS PRIORITY
 C.LIN: .BLKB 1 ; NUMBER OF LINES ASSIGNED TO THIS LLC
 C.DEV: .BLKW 1 ; PSEUDO DEVICE NAME FOR THIS LLC
 C.NLT: .BLKB 1 ; NUMBER OF LINE TABLE TO ALLOCATE
 C.EXT: .BLKB 1 ; PROCESS EXTENSION IN BLOCKS
 C.CTM: .BLKW 1 ; COUNTER TIMER
 CL.LEN=.
 .PSECT
 C.NLTD = 12 ; NLT FOR DDM/DLC PROCESSES
 C.NCTD = 13 ; NCT FOR DDM/DLC PROCESSES
 MSGDF$
69 000000
70
71 ;****
72 ; LOCAL DATA
73 ;****
74
75 000000 TEMP: .BLKW 1 ; ADDRESS OF LLC TEMPLATE
76 000002 132 106 056 ZLLC: .ASCII /ZF.LLC!/ ; MARK PROCESS AS AN LLC
77 000005 114 103
78 000010 041
79 000007 000007 ZLLCL=-ZLLC ; AND LENGTH
80 000011 132 106 056 ZMTM: .ASCII /ZF.MTM!/ ; MARK PROCESS AS NEEDING MULTIPLE TIMERS
81 000014 115 124 115
82 000017 041
83 000007 000007 ZMTML=-ZMTM ; AND LENGTH
84 000020 132 106 056 ZCOU: .ASCII /ZF.COU!/ ; MARK PROCESS AS SUPPORTING COUNTERS
85 000023 103 117 125
86 000026 041
87 000007 000007 ZCOUL=-ZCOU ; AND LENGTH
88 000027 132 106 056 ZX3P: .ASCII /ZF.X3P!/ ; MARK PROCESS AS NEEDING X.25 LEVEL 3
89 000032 130 063 120
90 000035 041

```

UPDLLC CREATED BY MACRO ON 29-JUN-85 AT 05:52 PAGE 1 N 11  
 SYMBOL CROSS REFERENCE CREF 04.00

| SYMBOL  | VALUE        | REFERENCES                                                 |
|---------|--------------|------------------------------------------------------------|
| CL.LEN  | = 000020     | #4-68                                                      |
| CS.LLC  | = ***** GX   | 6-200                                                      |
| CS.MOD  | = ***** GX   | 6-207                                                      |
| C.CTM   | = 000016     | #4-68 13-405                                               |
| C.DEV   | = 000012     | #4-68                                                      |
| C.EXT   | = 000015     | #4-68 12-385                                               |
| C.FLG   | = 000006     | #4-68 8-268 8-273 8-278 8-283 8-288 8-293 8-298            |
| C.LIN   | = 000011     | #4-68 10-344                                               |
| C.NAM   | = 000004     | #4-68 7-245                                                |
| C.NCTD  | = 000013     | #4-68                                                      |
| C.NLT   | = 000014     | #4-68 11-365                                               |
| C.NLTD  | = 000012     | #4-68                                                      |
| C.PRI   | = 000010     | #4-68 9-323                                                |
| C.STS   | = ***.*** GX | 6-200                                                      |
| F.NRBD  | = ***** GX   | *6-209                                                     |
| LLCKTB  | = 000000 RG  | #5-94 6-214                                                |
| LLCSTB  | = 000000 RG  | #5-94 6-215                                                |
| SETCTM  | = 000636 R   | #13-403                                                    |
| SETEXT  | = 000610 R   | #12-383                                                    |
| SETFLG  | = 000224 R   | #8-261                                                     |
| SETLIN  | = 000530 R   | #10-341                                                    |
| SETNAM  | = 000166 R   | #7-232                                                     |
| SETNLT  | = 000560 R   | #11-362                                                    |
| SETPRI  | = 000500 R   | #9-320                                                     |
| TEMP    | = 000000 R   | #4-75 *6-206 7-244 8-263 9-321 10-342 11-363 12-384 13-404 |
| ZCOU    | = 000020 R   | #4-80 4-81 8-285                                           |
| ZCOUL   | = 000007     | #4-81 8-286                                                |
| ZF.COU  | = ***** GX   | 8-283                                                      |
| ZF.INI  | = ***** GX   | 8-298                                                      |
| ZF.MFL  | = ***** GX   | 8-273                                                      |
| ZF.MTM  | = ***** GX   | 8-278                                                      |
| ZF.SLI  | = ***** GX   | 8-293                                                      |
| ZF.TIM  | = ***** GX   | 8-268                                                      |
| ZF.X3P  | = ***** GX   | 8-288                                                      |
| ZINI    | = 000045 R   | #4-86 4-87 8-300                                           |
| ZINIL   | = 000007     | #4-87 2-301                                                |
| ZLLC    | = 000002 R   | #4-76 4-77 8-265                                           |
| ZLLCL   | = 000007     | #4-77 8-266                                                |
| ZMTM    | = 000011 R   | #4-78 4-79 8-280                                           |
| ZMTML   | = 000007     | #4-79 8-281                                                |
| ZSLI    | = 000036 R   | #4-84 4-85 8-295                                           |
| ZSLIL   | = 000007     | #4-85 8-296                                                |
| ZX3P    | = 000027 R   | #4-82 4-83 8-290                                           |
| ZX3PL   | = 000007     | #4-83 8-291                                                |
| \$ALPHA | = 000022     | #5-94                                                      |
| \$ANY   | = 000020     | #5-94                                                      |
| \$BLANK | = 000006     | #5-94                                                      |
| \$BUFI  | = ***** GX   | 6-209                                                      |
| \$CAT5  | = ***** GX   | 7-243                                                      |
| \$CNVD  | = ***** GX   | 11-367 12-387 13-407                                       |
| \$CNVO  | = ***** GX   | 9-325 10-346                                               |
| \$DIGIT | = 000024     | #5-94                                                      |

UPDLOG      CREATED BY    MACRO    ON 29-JUN-85 AT 05:53      PAGE 2      N 12

MACRO CROSS REFERENCE      CREF    04.00

MACRO NAME      REFERENCES

|         |        |        |        |        |
|---------|--------|--------|--------|--------|
| DBGTP\$ | #6-116 |        |        |        |
| EPRINT  | #4-77  |        |        |        |
| ERROR\$ | #4-66  |        |        |        |
| ISTAT\$ | #4-90  | 6-116  |        |        |
| MTRAN\$ | #6-116 |        |        |        |
| PRINT   | #4-61  |        |        |        |
| STATE\$ | #4-90  | 6-121  | #6-124 | #6-127 |
| TRAN\$  | #4-90  | #6-122 | #6-125 |        |



UPDNOD      CREATED BY    MACRO    ON 29-JUN-85 AT 05:53      PAGE 1      N 13  
 SYMBOL CROSS REFERENCE      CREF    04.00

| SYMBOL    | VALUE       | REFERENCES                                                                                         |
|-----------|-------------|----------------------------------------------------------------------------------------------------|
| CS.MOD    | = ***** GX  | 6-183                                                                                              |
| CS.NOD    | = ***** GX  | 6-176                                                                                              |
| C.HOS     | = 000054    | #4-91      10-313      10-337                                                                      |
| C.LEN     | = 000056    | #4-92                                                                                              |
| C.NAM     | = 000004    | #4-86      7-219      7-222                                                                        |
| C.NID     | = 000010    | #4-88      8-242                                                                                   |
| C.NUM     | = 000052    | #4-90      9-266      9-290                                                                        |
| C.STS     | = ***** GX  | 6-176      6-183      9-264      10-311                                                            |
| F.NRBD    | = ***** GX  | *6-185                                                                                             |
| LA        | = 000074    | #4-75                                                                                              |
| NODKTB    | = 000000 RG | #5-101      6-190                                                                                  |
| NODSTB    | = 000000 RG | #5-101      6-191                                                                                  |
| NO\$EXA   | = ***** GX  | 9-264                                                                                              |
| NO\$HOA   | = ***** GX  | 10-311                                                                                             |
| RA        | = 000076    | #4-76                                                                                              |
| SETHOA    | = 000314 R  | #10-308                                                                                            |
| SETHOS    | = 000364 R  | #10-334                                                                                            |
| SETNAM    | = 000114 R  | #7-216                                                                                             |
| SETNID    | = 000156 R  | #8-239                                                                                             |
| SETNNA    | = 000212 R  | #5-261                                                                                             |
| SETNNM    | = 000262 R  | #9-287                                                                                             |
| TEMP      | = 000000 R  | #4-77      *6-182      7-218      7-221      8-241      9-263      9-289      10-310      10-336   |
| \$ALPHA   | = 000022    | #5-101                                                                                             |
| \$ANY     | = 000020    | #5-101                                                                                             |
| \$BLANK   | = 000006    | #5-101                                                                                             |
| \$BUFI    | = ***** GX  | 6-185                                                                                              |
| \$CNVD    | = ***** GX  | 9-271      10-318      10-339                                                                      |
| \$CSTA    | = ***** GX  | 7-220      7-223                                                                                   |
| \$DIGIT   | = 000024    | #5-101                                                                                             |
| \$DNUMB   | = 000014    | #5-101                                                                                             |
| \$EOS     | = 000012    | #5-101                                                                                             |
| \$ERROR   | = ***** GX  | *6-194                                                                                             |
| \$EXIT    | = 000000    | #5-101                                                                                             |
| \$FAIL    | = 177777    | #5-101                                                                                             |
| \$GPRM    | = *****     | 5-101                                                                                              |
| \$LAMDA   | = 000000    | #5-101                                                                                             |
| \$MOVST   | = ***** GX  | 6-187                                                                                              |
| \$MVASC   | = ***** GX  | 8-244                                                                                              |
| \$NUMPR   | = 000002    | #5-101                                                                                             |
| \$OBUFI   | = ***** GX  | 7-217      *7-224      8-240      *8-246      9-262      *9-272      9-288      *9-293      10-309 |
|           |             | *10-319      10-335      *10-340                                                                   |
| \$OFDB    | = ***** GX  | *6-185                                                                                             |
| \$RAD50   | = 000016    | #5-101                                                                                             |
| \$RONLY   | = *****     | 5-101      5-101                                                                                   |
| \$STRNG   | = 000004    | #5-101                                                                                             |
| \$SUBXP   | = 000010    | #5-101                                                                                             |
| \$TMLST   | = ***** GX  | 6-172                                                                                              |
| \$UPNOD   | = 000002 RG | #6-171                                                                                             |
| \$\$\$FLG | = 177777    | #5-101                                                                                             |
| \$\$\$KEY | = 177777    | #5-101                                                                                             |
| CONER     | = ***** GX  | 6-200                                                                                              |
| TPARS     | = ***** GX  | 6-192                                                                                              |

|                  |            |          |              |          |          |          |              |          |
|------------------|------------|----------|--------------|----------|----------|----------|--------------|----------|
| 000027R          | OBJDEF     | 000002R  | SETYPE       | 000326R  | \$EOS =  | 000012   | \$RAD50=     | 000016   |
| CS.OBJ= ***** GX | OBJDF      | 000017R  | TEMP         | 000000R  | \$ERROR= | ***** GX | \$STRNG=     | 000004   |
| C.CPY 000012     | OBJKT9     | 000000RG | 003 \$ALPHA= | 000022   | \$EXIT = | 000000   | \$SUBXP=     | 000010   |
| C.FLG 000005     | OBJLEN=    | 000014   | \$ANY =      | 000020   | \$FAIL = | 177777   | \$TMLST=     | ***** GX |
| C.LEN = 000013   | OBJLN =    | 000010   | \$BLANK=     | 000006   | \$LAMD=  | 000000   | \$UPDOBJ     | 000150RG |
| C.NAM = 000006   | OBJSTB     | 000000RG | 002 \$BUFI = | ***** GX | \$MOVPT= | ***** GX | \$WDATA=     | ***** GX |
| C.STS = ***** GX | RA =       | 000076   | \$CNVD =     | ***** GX | \$MVASC= | ***** GX | \$\$\$\$FLG= | 177777   |
| C.TYP 000004     | SETCPY     | 000534R  | \$CNVQ =     | ***** GX | \$NUMBR= | 000002   | \$\$\$\$KEY= | 177777   |
| EXIT 000044R     | 002 SETFLG | 000500R  | \$CSTA =     | ***** GX | \$OBUF = | ***** GX | \$\$\$\$STA= | 000000   |
| F.NRBD= ***** GX | SETNAM     | 000356R  | \$DIGIT=     | 000024   | \$OFDB = | ***** GX | .TPARS=      | ***** GX |
| LA = 000074      | SETVfy     | 000444R  | \$DNUMB=     | 000014   |          |          |              |          |

. ABS. 000013 000 (RW,I,GBL,ABS,OVR)  
000562 001 (RW,I,LCL,REL,CON)  
\$STATE 000050 002 (RW,D,LCL,REL,CON)  
\$KTAB 000000 003 (RW,D,LCL,REL,CON)  
\$KSTR 000000 004 (RW,D,LCL,REL,CON)  
Errors detected: 0

\*\*\* Assembler statistics

Work file reads: 0  
Work file writes: 0  
Size of work file: 10440 Words ( 41 Pages)  
Size of core pool: 14440 Words ( 55 Pages)  
Operating system: RSX-11M/PLUS

Elapsed time: 00:00:19.23  
SY:UPDOBJ.V2,[132,134]UPDOBJ/CR/-SP=SY:[1,1]RSXMCML/ML,[130,110]NETLIB/ML,[132,10]UPDOBJ

```

240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255 000240
256 000240 016700 000000G
257 000244 016701 000000G
258 000250 020127 177700
259 000254 101077 000000G
260 000256 020067 000000G
261 000262 103002
262 000264 016701 000000G
263 000270 062701 000077
264 000274 000241
265 000276 006001
266
267 000005
268
269
270
271 000312 004767 000000G
272 000316 010067 000000G
273 000322 000207

```

```

.SBTTL SETALL - SET THE BYTE-AREA EXTENSION
*** - SETALL - SET THE POOL BYTE-AREA EXTENSION

INPUT:
$OBUF = CURRENT POSITION IN OUTPUT BUFFER
$BYTXT = CURRENT BYTE EXTENSION IN BYTES
$BYTMN = MINIMUM BYTE EXTENSION IN BYTES

OUTPUT:
BYTE-AREA EXTENSION VALUE IS STORED IN THE OUTPUT BUFFER
$OBUF IS UPDATED

SETALL:
MOV $OBUF,R0 ; GET CURRENT POSITION IN OUTPUT BUFFER
MOV $BYTXT,R1 ; GET CURRENT BYTE EXTENSION
CMP R1,#<1023.*64.> ; COMPARE WITH MAXIMUM
BHI 5$; IF H1, OVERFLOW - USE MINIMUM INSTEAD
CMP R1,$BYTMN ; COMPARE WITH MINIMUM
BHS 10$; IF H1, GO WITH IT
5$: MOV $BYTMN,R1 ; ELSE USE MINIMUM VALUE
10$: ADD #63.,R1 ; ALWAYS ROUND UP TO NEXT BLOCK BOUNDARY
CLC ; DIVIDE BY 64. FOR NUMBER OF BLOCKS (UNSIGNED)
ROR R1 ; ...

.REPT 5
ASR R1 ; ...
.ENDR

CALL $CNVD ; STORE NUMBER IN OUTPUT BUFFER
MOV R0,$OBUF ; UPDATE CURRENT POSITION IN OUTPUT BUFFER
RETURN

```

[illegible]

|     |        |
|-----|--------|
| 5   | UPDDSC |
| 6   |        |
| 7   |        |
| 8   |        |
| 9   |        |
| 10  |        |
| 11  |        |
| 12  |        |
| 13  |        |
| 14  |        |
| 15  |        |
| 16  |        |
| 17  |        |
| 18  |        |
| 19  |        |
| 20  |        |
| 21  |        |
| 22  |        |
| 23  |        |
| 24  |        |
| 25  |        |
| 26  |        |
| 27  |        |
| 28  |        |
| 29  |        |
| 30  |        |
| 31  |        |
| 32  |        |
| 33  |        |
| 34  |        |
| 35  |        |
| 36  |        |
| 37  |        |
| 38  |        |
| 39  |        |
| 40  |        |
| 41  |        |
| 42  |        |
| 43  |        |
| 44  |        |
| 45  |        |
| 46  |        |
| 47  |        |
| 48  |        |
| 49  |        |
| 50  |        |
| 51  |        |
| 52  |        |
| 53  |        |
| 54  |        |
| 55  |        |
| 56  |        |
| 57  |        |
| 58  |        |
| 59  |        |
| 60  |        |
| 61  |        |
| 62  |        |
| 63  |        |
| 64  |        |
| 65  |        |
| 66  |        |
| 67  |        |
| 68  |        |
| 69  |        |
| 70  |        |
| 71  |        |
| 72  |        |
| 73  |        |
| 74  |        |
| 75  |        |
| 76  |        |
| 77  |        |
| 78  |        |
| 79  |        |
| 80  |        |
| 81  |        |
| 82  |        |
| 83  |        |
| 84  |        |
| 85  |        |
| 86  |        |
| 87  |        |
| 88  |        |
| 89  |        |
| 90  |        |
| 91  |        |
| 92  |        |
| 93  |        |
| 94  |        |
| 95  |        |
| 96  |        |
| 97  |        |
| 98  |        |
| 99  |        |
| 100 |        |
| 101 |        |
| 102 |        |
| 103 |        |
| 104 |        |
| 105 |        |
| 106 |        |
| 107 |        |
| 108 |        |
| 109 |        |
| 110 |        |
| 111 |        |
| 112 |        |
| 113 |        |
| 114 |        |
| 115 |        |
| 116 |        |
| 117 |        |
| 118 |        |
| 119 |        |
| 120 |        |
| 121 |        |
| 122 |        |
| 123 |        |
| 124 |        |
| 125 |        |
| 126 |        |
| 127 |        |
| 128 |        |
| 129 |        |
| 130 |        |
| 131 |        |
| 132 |        |
| 133 |        |
| 134 |        |
| 135 |        |
| 136 |        |
| 137 |        |
| 138 |        |
| 139 |        |
| 140 |        |
| 141 |        |
| 142 |        |
| 143 |        |
| 144 |        |
| 145 |        |
| 146 |        |
| 147 |        |
| 148 |        |
| 149 |        |
| 150 |        |
| 151 |        |
| 152 |        |
| 153 |        |
| 154 |        |
| 155 |        |
| 156 |        |
| 157 |        |
| 158 |        |
| 159 |        |
| 160 |        |
| 161 |        |
| 162 |        |
| 163 |        |
| 164 |        |
| 165 |        |
| 166 |        |
| 167 |        |
| 168 |        |
| 169 |        |
| 170 |        |
| 171 |        |
| 172 |        |
| 173 |        |
| 174 |        |
| 175 |        |
| 176 |        |
| 177 |        |
| 178 |        |
| 179 |        |
| 180 |        |
| 181 |        |
| 182 |        |
| 183 |        |
| 184 |        |
| 185 |        |
| 186 |        |
| 187 |        |
| 188 |        |
| 189 |        |
| 190 |        |
| 191 |        |
| 192 |        |
| 193 |        |
| 194 |        |
| 195 |        |
| 196 |        |
| 197 |        |
| 198 |        |
| 199 |        |
| 200 |        |
| 201 |        |
| 202 |        |
| 203 |        |
| 204 |        |
| 205 |        |
| 206 |        |
| 207 |        |
| 208 |        |
| 209 |        |
| 210 |        |
| 211 |        |
| 212 |        |
| 213 |        |
| 214 |        |
| 215 |        |
| 216 |        |
| 217 |        |
| 218 |        |
| 219 |        |
| 220 |        |
| 221 |        |
| 222 |        |
| 223 |        |
| 224 |        |
| 225 |        |
| 226 |        |
| 227 |        |
| 228 |        |
| 229 |        |
| 230 |        |
| 231 |        |
| 232 |        |
| 233 |        |
| 234 |        |
| 235 |        |
| 236 |        |
| 237 |        |
| 238 |        |
| 239 |        |
| 240 |        |
| 241 |        |
| 242 |        |
| 243 |        |
| 244 |        |
| 245 |        |
| 246 |        |
| 247 |        |
| 248 |        |
| 249 |        |
| 250 |        |
|     |        |

|   |    |        |
|---|----|--------|
| E | 10 |        |
| F | 10 |        |
| G | 10 |        |
| H | 10 |        |
| I | 10 |        |
| J | 10 |        |
| K | 10 | UPDLLC |
| L | 10 |        |
| M | 10 |        |
| N | 11 |        |
| B | 11 |        |
| C | 11 |        |
| D | 11 |        |
| E | 11 |        |
| F | 11 |        |
| G | 11 |        |
| H | 11 |        |
| I | 11 |        |
| J | 11 |        |
| K | 11 |        |
| L | 11 |        |
| M | 11 |        |
| N | 11 |        |
| B | 12 |        |
| C | 12 |        |
| D | 12 | UPDL0G |
| E | 12 |        |
| F | 12 |        |
| G | 12 |        |
| H | 12 |        |
| I | 12 |        |
| J | 12 |        |
| K | 12 |        |
| L | 12 |        |
| M | 12 |        |
| N | 13 | UPDN0D |
| B | 13 |        |
| C | 13 |        |
| D | 13 |        |
| E | 13 |        |
| F | 13 |        |
| G | 13 |        |
| H | 13 |        |
| I | 13 |        |
| J | 13 |        |
| K | 13 |        |
| L | 13 |        |
| M | 13 |        |
| N | 13 |        |
| B | 14 | UPD0BJ |
| C | 14 |        |
| D | 14 |        |
| E | 14 |        |
| F | 14 |        |
| G | 14 |        |
| H | 14 |        |
| I | 14 |        |
| J | 14 |        |
| K | 14 |        |
| L | 14 |        |

| Protein | Residue | Score |
|---------|---------|-------|
| UPDPA   | M       | 14    |
|         | N       | 14    |
|         | B       | 13    |
|         | C       | 13    |
|         | D       | 13    |
|         | E       | 13    |
|         | F       | 15    |
|         | G       | 15    |
|         | H       | 15    |
|         | I       | 15    |
| UPDPAR  | J       | 15    |
|         | K       | 15    |
|         | L       | 13    |
|         | M       | 13    |
|         | N       | 15    |
|         | B       | 15    |
|         | C       | 16    |
|         | D       | 16    |
|         | E       | 16    |
|         | F       | 16    |
| UPDPSN  | G       | 16    |
|         | H       | 16    |
|         | I       | 16    |
|         | J       | 16    |
|         | K       | 16    |
|         | L       | 16    |
|         | M       | 16    |